

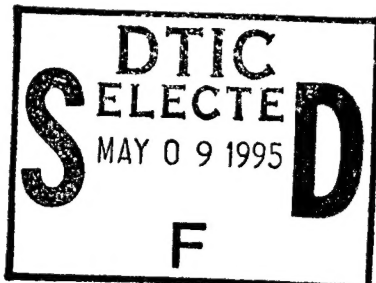
# STSC

---

## Software Engineering Environment Technology Report

*April 1994*

---



*Software Engineering Environment Domain*

*Robert Hanrahan*

*Charles Daud*

*Kenneth Meiser*

*Judi Peterson*

This document has been approved  
for public release and sale; its  
distribution is unlimited.


19950505 214

*Software Technology Support Center*

This technical report was prepared by the Software Technology Support Center, OO-ALC/TISE, Hill AFB, Utah.

The ideas and findings expressed in this report should not be construed as an official position of the United States Air Force. It is published in the interest of scientific and technical information exchange.

Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification .....	
By <i>Karen Rasmussen</i>	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
<i>A-1</i>	



This document is available through the Software Technology Support Center (STSC). To obtain a copy, contact STSC Customer Service, OO-ALC/TISE, 7278 Fourth Street, Hill AFB, UT 84056-5205; Voice: 801-777-7703, DSN 458-7703, Fax: 801-777-8069, e-mail: mcdonalb@stsc.hill.af.mil.

## **Preface**

Software Engineering Environment (SEE) technology is one of several software product domains being investigated by the U. S. Air Force's Software Technology Support Center (STSC) at the Ogden Air Logistics Center, Hill Air Force Base, Utah. This report is produced by the STSC to increase awareness and understanding of SEE technology. The information in this report is aimed at Air Force managers and technical people who must make the decisions about acquiring SEE technology and who must prepare their organizations to employ it effectively; use of this report should be the first step in that process.

This report examines the software challenges facing today's Air Force and the role of SEE technology in meeting these challenges. The concepts of SEE technology are introduced and explained. The report examines current SEE technology and provides information about specific products in the marketplace. Finally, the report addresses future directions in SEE technology as an aid in planning long-range strategies.

It is assumed that readers of this report are familiar with Department of Defense and Air Force software policies and practices.

## Table of Contents

<b>Preface .....</b>	<b>iii</b>
<b>1 Introduction.....</b>	<b>1-1</b>
1.1 Purpose of Report.....	1-1
1.2 Intended Audience of Report .....	1-1
1.3 Software Challenges .....	1-1
1.3.1 Improving software quality and productivity.....	1-2
1.3.2 Providing affordable weapon system flexibility.....	1-2
1.3.3 Ways in which the Air Force is meeting the software challenges	1-3
1.4 SEE Technology .....	1-4
<b>2 SEE Technology Tutorial.....</b>	<b>2-1</b>
2.1 Purpose and Objectives of SEE Technology.....	2-1
2.1.1 Management issues.....	2-1
2.1.2 Benefits .....	2-2
2.2 Concepts .....	2-2
2.2.1 Software Engineering Environment .....	2-2
2.2.2 Services .....	2-3
2.2.3 Repository.....	2-4
2.2.4 Framework .....	2-5
2.2.5 Tools.....	2-5
2.2.6 Integration.....	2-6
2.2.7 Open versus closed SEE .....	2-7
2.3 Historical Background.....	2-7
2.3.1 Computer-aided software engineering (CASE) .....	2-8
2.3.2 Integrated project support environment (IPSE).....	2-8
2.3.3 CASE coalition .....	2-8
<b>3 Specific Considerations .....</b>	<b>3-1</b>
3.1 Reference Models (RMs) .....	3-1
3.1.1 NIST/ECMA Frameworks RM.....	3-1
3.1.2 RM for a Project Support Environment (PSE). .....	3-1
3.2 Standards .....	3-1
3.2.1 General Information .....	3-1
3.2.2 Conclusions.....	3-2
<b>4 State of SEE Technology.....</b>	<b>4-1</b>
4.1 Summary of State-of-the-Practice .....	4-1
4.2 Summary of the State-of-the-Art.....	4-2
4.3 Government Initiatives .....	4-2

4.3.1	The Process-Oriented Software Life Cycle Support Environment (ProSLCSE) .....	4-3
4.3.2	Software Technology for Adaptable Reliable Systems (STARS) Program .....	4-3
4.3.3	Integrated Computer-Aided Software Engineering (I-CASE)...	4-3
4.3.4	The Software Technology Support Center (STSC) .....	4-3
4.3.5	The Integrated SEE Special Interest Group (ISEE SIG) .....	4-4
4.3.6	The Software Engineering Institute (SEI).....	4-4
4.3.7	The Project Support Environment Standards Working Group (PSESWG) .....	4-4
4.3.8	Corporate Information Management (CIM) Program.....	4-4
4.4	Future Directions and Trends .....	4-5
4.4.1	Software development process.....	4-5
4.4.2	Reuse.....	4-6
4.4.3	Security.....	4-6
5	Applications.....	5-1
5.1	Why Do You Use SEE Technology?.....	5-1
5.2	When Do You Use SEE Technology?.....	5-1
5.3	How Do You Use SEE Technology?.....	5-2
5.4	Important Issues .....	5-3
5.5	Management and Business Matters .....	5-3
5.6	Technical Hints .....	5-4
5.6.1	Organizational needs.....	5-4
5.6.2	Product requirements .....	5-5
5.6.3	Vendor evaluation .....	5-5
5.7	Acquisition of a SEE .....	5-6
6	SEE Technology Products.....	6-1
6.1	Scope.....	6-1
6.2	The Information Gathering Process .....	6-1
6.3	Product Lists .....	6-1
6.4	Product Sheets/Vendor Questionnaires .....	6-2
6.5	Product Critiques .....	6-2
6.6	Criteria.....	6-2
7	Conclusions.....	7-1

## List of Appendices

Appendix A Acronyms and Terms

Appendix B Glossary

Appendix C SEE Products List by Source

<b>Appendix D</b>	<b>SEE Product Sheets</b>
<b>Appendix E</b>	<b>SEE Product Critiques</b>
<b>Appendix F</b>	<b>SEE Product Features Matrix</b>
<b>Appendix G</b>	<b>Standards</b>
<b>Appendix H</b>	<b>Bibliography</b>
<b>Appendix I</b>	<b>Recommended Reading</b>

## **List of Figures**

<b>Figure 2-1: Framework Services .....</b>	<b>2-4</b>
<b>Figure 2-2: Conceptual Model vs. Actual Model .....</b>	<b>2-6</b>

## **List of Tables**

<b>SEE Products List by Source .....</b>	<b>C-2</b>
<b>SEE Features Matrix .....</b>	<b>F-7</b>

# **1 Introduction**

This section states the purpose of the report, identifies the intended audience, and discusses the software challenges facing the Air Force.

## **1.1 Purpose of Report**

The purpose of this report is to impart a basic understanding of Software Engineering Environment (SEE) technology, give an update of its current state, recommend guidelines on when and how to use it, and provide information on its major products. SEE technology is one of several technology domains studied by the Software Technology Support Center (STSC) in fulfillment of its mission to "assist Air Force software organizations to identify, evaluate, and adopt technologies that will improve the quality of their software products, their efficiency in producing them, and their ability to accurately predict the cost and schedule of their delivery."<sup>1</sup> The STSC's mission is carried out using technology evaluation, information exchange, and technology insertion. This report is the primary means by which the STSC shares the information it is gathering through its on-going evaluation of SEE technology.

## **1.2 Intended Audience of Report**

Managers and others who are responsible for choosing, recommending, or guiding the implementation of technologies and products for the development and post-deployment support of software within the Air Force are the intended audience for this report.

## **1.3 Software Challenges**

Improving software quality and productivity is a major challenge faced by the Air Force, along with the rest of the Department of Defense (DoD) and the nonmilitary community as well. In addition, providing affordable weapon system flexibility through software is

---

<sup>1</sup> STSC mission statement, 8 December 1993

a specific challenge the Air Force accepts as a member of the United States military establishment.

### **1.3.1 Improving software quality and productivity**

Improving software quality is basically to improve how well software meets the requirements and expectations of the users. It also means ensuring that software is adequate, reliable, and efficient. Improving productivity means favorably increasing the ratio between the resources required to develop software and the size and complexity of the developed software.

The growth in computer use and computer hardware capabilities has placed demands of increasing magnitude and complexity on computer software. Software development processes, along with the attendant methodologies, which may have worked well in the past, often break down when applied to the development of today's software. For example, studies show that every five years the sizes of software projects, as measured by source lines of code (SLOC), increase by an order of magnitude, and that the scaling of the development effort demanded by the order-of-magnitude increases, requires fundamental changes in the development process. As the sizes of software projects have increased, software development processes based on individual programmers have given way to processes based on small teams and, in turn, small teams have given way to larger teams. Scaling up software development processes by merely increasing team sizes reaches limits on effective project management and resources availability [CURTIS].

Today's users of software, demand software applications of greater size and complexity than before. Advances in computer hardware capabilities are more than adequate to match the demands of users; however, software, as it is developed using prevalent processes and methodologies, is not. The challenge is finding software development processes with attendant methodologies and technologies that meet user demands and that improve software quality and productivity.

### **1.3.2 Providing affordable weapon system flexibility**

The military, like the business world today, sees software providing the versatility and leverage to achieve its performance goals. For example, software demonstrated its flexibility to quickly change weapon system capabilities in Desert Storm, the most newsworthy being the development of a new software package for the Patriot Missile system to counter the SCUD. Because of the versatility and leverage provided by

software, the DoD's appetite for software in the future has been described as "virtually insatiable" [MOSEMANN 93].

In today's world of shrinking budgets, providing affordable, flexible software systems requires cost control and predictability that are not found in the prevalent, traditional software development processes. Increasingly, the Air Force demands that software be developed within predictable costs and schedule.

### **1.3.3 Ways in which the Air Force is meeting the software challenges**

#### **(1) Process Improvement**

The Air Force is committed to a policy of software development process improvement. Through process improvement, the Air Force expects to overcome the limitations inherent in the traditional process models used for software development. The Software Engineering Institute (SEI), as a Federally Funded Research and Development Center (FFRDC) sponsored by the DoD, has recognized that process capability is built in progressive stages, which the SEI calls *maturity levels*. Each maturity level provides a foundation for the improvements needed to reach the next level. The SEI has formalized this concept of maturity levels for software organizations in a model called the *Capability Maturity Model (CMM)* [MCKEEHAN 93]. Currently, the Air Force's process improvement policy is based on using the CMM to raise the maturity level of its software organizations.

Another way that the Air Force facilitates its process improvement policy is by establishing Software Engineering Process Groups (SEPGs) and providing training to its organizations in the management of technological change.

#### **(2) Technology Insertion**

The Air Force uses technology insertion as a means to improve software quality and productivity within its organizations, especially technology that provides the leverage to overcome the limitations of managing complex projects within budget constraints. The Air Force established the STSC to advance this kind of technology insertion. Additionally, software technology insertion has been facilitated by the Air Force through the funding of pilot projects for the evaluation and promotion of technologies.

(3) Other Policies

Other software policies of note are the mandating of DoD-approved higher-order languages for software development, the most well known being Ada; encouraging the reuse of existing software and designing new software for reuse; and requiring the implementation of effective software measurement in the process of managing software development.

#### **1.4 SEE Technology**

The DoD has looked to SEE technology as a means to improve software quality and software development productivity, and to enable the predictability in costs and schedule required to develop and maintain complex software systems. SEE technology supports and facilitates the software development process and post-deployment software support through the automation of software engineering practices.

SEEs provide the infrastructure and means to integrate the software tools that assist the system analyst, software designer, programmer, configuration manager, software tester, quality assurance engineer, project manager, reuse engineer, and so on, with the development, management, and maintenance of software throughout its life cycle. SEEs collect, store, and make available the artifacts of software engineering, such as requirements, design documents, code, test plans, metrics, schedules, and resource allocations. SEEs provide communications between team members, and between the technical and management. SEEs promote common user interfaces among software tools, reduce redundant information, and facilitate coordination of development activities.

## **2 SEE Technology Tutorial**

This section introduces the concepts and background needed for a basic understanding of SEE technology. In this section, the purpose and objectives of SEE technology, as well as related management issues and benefits, are presented. Some of the information in this section is based on work done by the U. S. Department of Commerce's National Institute of Standards and Technology (NIST), the U. S. Navy's Next Generation Computer Resources (NGCR) Program, and the Software Engineering Institute (SEI).

### **2.1 Purpose and Objectives of SEE Technology**

The purpose of SEE technology is to provide automated support to the software development and post-development processes, especially for large-scale or complex applications. Objectives for using SEE technology are to improve software quality and productivity, and to enhance predictability and, in turn, reduce the risk associated with software development.

#### **2.1.1 Management issues**

Software development and post-deployment software support, especially for large-scale and complex systems, are labor-intensive efforts that require high skill levels, above average intellect, and creative talent. Traditional software practices tend to squander these labor resources through inadequate control, miscommunications, rework, and lack of relief from noncreative tasks. Traditional practices often deny the project manager the information needed to effectively use labor resources, i.e., efficiency, to meet cost and schedule projections, i.e., predictability, or to establish realistic projections in the first place.

SEE technology facilitates project management by making timely and complete project information visible to the project manager [BROWN 92]. This visibility gives the manager information on the actual development status allowing the manager to more effectively utilize available resources which, in turn, reduces the risk of not meeting the cost and schedule goals of the project. Project information captured in this way improves the predictability and repeatability of the software development processes. In a talk at the 1993 Software Technology Conference (STC) the Deputy Assistant Secretary

(Communications, Computers and Logistics) of the Air Force stated that *predictability* is what the Pentagon wants in its software processes [MOSEMANN 93].

### **2.1.2 Benefits**

When looking at the benefits of SEE technology, it must be kept in mind that the application of SEE technology, in and of itself, is not a "silver bullet"; the benefits are realized by the organization that has effective software processes defined and enacted. Overall, effective use of SEE technology better an organization's software quality and productivity by enhancing the organization's ability to produce and deliver software, within budget and with fewer problems, that meets users' requirements. Continued use of SEE technology achieves greater predictability and repeatability from an organization's software development and post-development processes through tool configurations designed for the work flow process, through automating parts of the process, and through the capture of useful metrics [CUTHILL 94].

The following potential benefits also are derived from SEE technology through the integration of tools across the software life cycle:

- Integration of requirements definition, design, and testing tools to support automatic test case generation
- Streamlining the traceability of requirements through design, coding, testing, and documentation preparation
- Developing consistent versions of requirements, design, code, documentation, and testing products
- Automatic reformatting and translation of products from one tool to another
- Process support from tools designed to encourage adherence to process guideline.

[CUTHILL 94]

## **2.2 Concepts**

The following paragraphs introduce and explain concepts that are essential to understanding SEE technology.

### **2.2.1 Software Engineering Environment**

A *software engineering environment* is a collection of software components working together for the purpose of providing automated support of software engineering

activities. A SEE facilitates the development and post-deployment support, i.e., maintenance, of software and the management of the software process.

### 2.2.2 Services

As an aid in classifying the functions performed by a SEE, the concept of *services* was created; essentially, a service is a function performed by a SEE. The services of a SEE are assigned to one of two classifications: (1) the services used directly by the end-user of the SEE, called the *end-user services*; and (2) the services that underlie and support the end-user services, called the *framework services* [NIST 93].

End-user services are the functions that directly support the execution of a software project. End-user services are also referred to as application development, engineering, or project life cycle services [NIST 93]. These services are grouped into the following categories:

- *Technical Engineering* services that "support activities related to the specification, implementation, and maintenance of systems"
- *Technical Management* services that support configuration management, change management, information management, reuse management, and metrics
- *Project Management* services that "support activities related to planning and executing a project." Included in these services are estimating, risk analysis, and project tracking
- *Support* services that "include services used by all users," such as text processing, publishing, electronic mail (e-mail), and bulletin board.

[NGCR 93b]

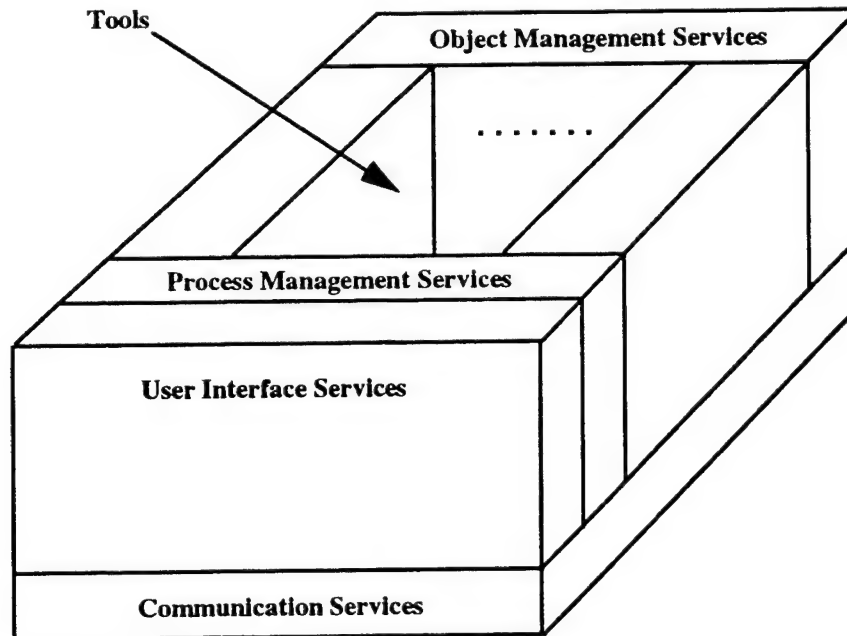
Framework services underlie and support the end-user services. Framework services are grouped into the following categories:

- *Object Management* services, the purpose of which "is the definition, storage, maintenance, management, and access of object entities and the relationships among them"
- *Process Management* services that provide facilities for process capture and enforcement
- *Communication* service that provides a standard communication mechanism that can be used between the software components of the SEE
- *Operating System* services that "include those services that are usually considered part of an operating system or executive"

- *User Interface* services that provide for interaction with other services in a consistent manner
- *Policy Enforcement* services that provide for user identification and authentication and other activities related to security.

[NIST 93]

Figure 2-1 (often referred to as the "toaster model") is an aid in conceptualizing the groupings of the framework services; it is not to be construed as an architecture of a SEE framework. The Policy Enforcement services grouping is not shown in the figure. The "tools", indicated in the figure, encompass the end-user services that are supported by the framework.



**Figure 2-1. Framework Services [NIST 93].**

### 2.2.3 Repository

The *repository* is a mechanism for storing, organizing, and making available all the information concerning a software system and project [MCCLURE 89]. Information in the repository typically covers the following topics:

- the software under development—requirements, design specifications and data, source code, test data, and project plans
- project resources

- organization policy, standards, and guidelines on the production of software
- post-development modifications [NIST 93].

A repository can supply many of the framework services listed in paragraph 2.2.2, especially the Object Management services [CUTHILL 94].

For a SEE to be effective, the repository must be shared and accessible by all the services. Having a shared repository is necessary to derive the maximum benefits from reusability, process management, and configuration control.

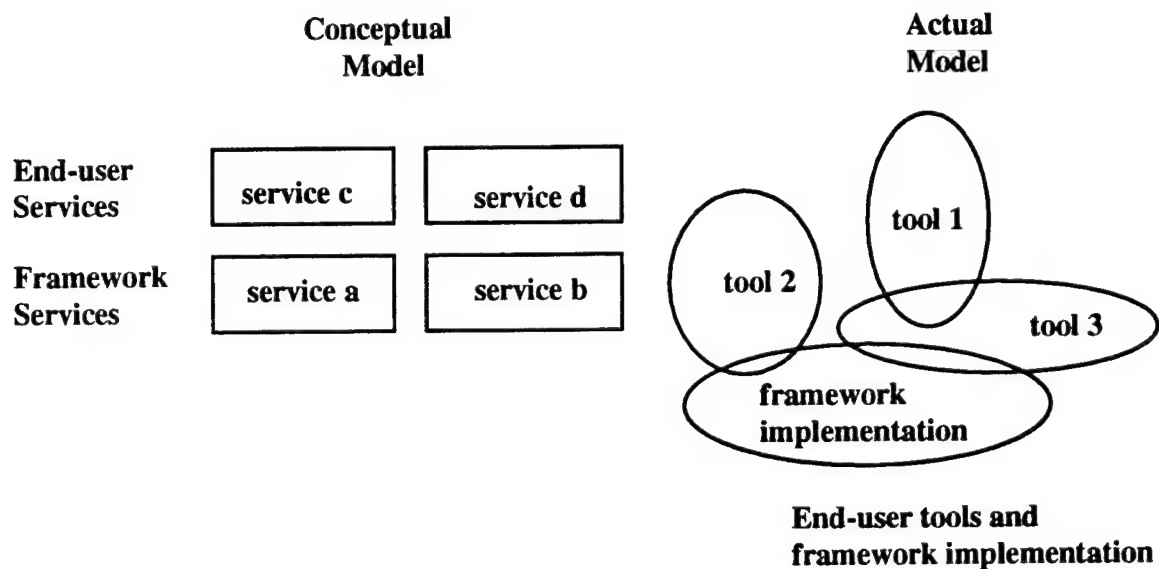
Terms often used synonymously with repository are *database*, *data dictionary*, and *encyclopedia*. These terms fail to capture the full meaning of a repository. A database is a means by which a repository is implemented. Data dictionaries and encyclopedias are usually associated more with passive storage of data; a repository is considered to be active storage.

#### **2.2.4 Framework**

The *framework* is the kernel of the SEE; it provides the framework services described in paragraph 2.2.2. In addition, the framework includes services for the administration of the framework, interfaces to the end-user services, and mechanisms for integrating the end-user services into the SEE [BROWN 92]. The *framework implementation* is the actual software product, or products, that provide the framework services within a specific SEE. Framework services are a conceptual model that has proven useful in identifying and categorizing the capabilities expected in a framework. A framework implementation is the actual model of the software products selected to provide the particular framework services desired in a SEE (see Figure 2-2).

#### **2.2.5 Tools**

*Tools* are the actual software components that provide the desired end-user services within an implementation of a SEE. Analogous to the conceptual model and the actual model discussed in paragraph 2.2.4, end-user services belong to the conceptual model and tools to the actual model (see Figure 2-2).



**Figure 2-2. Conceptual Model vs. Actual Model [PSESWG 93].**

### 2.2.6 Integration

Basically, *integration* is the process of combining software components, hardware components, or both, into an overall system [IEEE 90]. Integration is what transforms a collection of unrelated software tools into a SEE. Generally, the mechanisms for doing this are:

- *data integration*—the sharing of information throughout the SEE, such as through a repository shared by all software components
- *control integration*—the synchronization of activities within a SEE
- *presentation integration*—providing common look and feel to the user interacting with the SEE
- *process integration*—the accessing of SEE functions based on a defined and enactable development process
- *framework integration*—the degree to which the tools make effective use of the framework [NIST 93].

In a SEE implementation, one or more of the integration mechanisms will be present at some level. An appropriate level of integration can increase the productivity of SEE users, for example, by reducing their training time through the use of presentation integration or by eliminating the burden of handling data in redundant files through data integration. One's perspective, whether a SEE user, SEE builder, or tool builder, colors what is expected or desired in the way of integration. SEE users are concerned with their

interaction with the SEE—the touch and look, the ease with which tasks can be performed, and the degree to which the SEE functions as one consistent tool. The SEE builder and tool builder desires a consistent interface for building tools and is concerned with how the SEE is structured, i.e., its architecture, and how its components relate to each other [NIST 93].

### **2.2.7 Open versus closed SEE**

A *closed* SEE, also referred to as a *closed environment*, is an environment that is provided essentially as a "black box," usually (but not necessarily) by one vendor. Consequently, tailoring of the environment by the customer with tools and framework products other than the vendor's is either not permitted or requires significant effort and cost to do. By contrast, an *open* SEE, also referred to as an *open environment*, allows the intermixing of tools and framework products from more than one vendor. Both types of environments have advantages and disadvantages.

Closed environments usually exhibit a high level of integration for the user and represent much less effort to install than an open environment. Experience shows that it is costly for a single vendor to design and build an entire SEE, including the upgrades and changes that are endemic of software products; consequently, the customer assumes a great deal of risk by being locked to a single vendor, especially as SEE technology continues to emerge and evolve.

The Federal government, including DoD, favors open environments; most important, the DoD does not see its best interests being served by dependence on a single supplier. Open environments allow tailoring of the SEE's functionality to match the needs of the project and the flexibility to use various methodologies. Adversely, open environments place greater demands than closed environments on the planning for and the procurement and installation of a SEE within an organization. Integration of an open environment is usually harder to achieve than a closed one.

## **2.3 Historical Background**

This subsection describes approaches that have been taken historically to automate the software development process and that have been instrumental in the evolution to SEE technology.

### **2.3.1 Computer-aided software engineering (CASE)**

*CASE* is defined to be the use of computers to aid in the software engineering process [IEEE 90]. Efforts in the early 1980s to automate software development documentation and diagramming, as well as the rise of the personal workstation for developing software, led to the emergence of CASE technology [MCCLURE 89]. As CASE tools proliferated, a differentiation into *upper CASE* tools and *lower CASE* tools appeared. Upper CASE tools support the requirements analysis and high-level design phases of the software life cycle; lower CASE tools support the software testing phase of the life cycle and also have been referred to as Computer-Aided Software Testing (CAST). By the mid-1980s, implementations of CASE tools usually had a design database and extensive graphic capabilities.

### **2.3.2 Integrated project support environment (IPSE)**

*IPSE* is an early SEE model that evolved in the 1980s from the concept of an Ada programming support environment (APSE). An IPSE is characterized by the use of data integration as a strategy for incorporating tools. An IPSE uses a data repository and an attendant data sharing mechanism for the integration of tools. The IPSE model of a SEE has found widespread acceptance with large-scale software developers.

### **2.3.3 CASE coalition**

*CASE coalition* is a type of SEE that appeared in the same time frame as the IPSE model and is characterized by the use of control integration as a strategy for incorporating tools. CASE coalition has proved popular among CASE tool vendors due to the low cost of integrating their products into the environment and has led to alliances among the vendors.

### **3 Specific Considerations**

In this section, the focus is on reference models and standards that apply to SEE technology.

#### **3.1 Reference Models (RMs)**

RMs were developed as a conceptual aid for examining standards and products [CUTHILL 94]. Two RMs important to SEEs are identified in the following paragraphs.

##### **3.1.1 NIST/ECMA Frameworks RM**

This RM was developed by the European Computer Manufacturers Association (ECMA, pronounced "ek-ma") and was modified by the National Institute of Standards and Technology (NIST) Integrated Software Engineering Environment (ISEE) Working Group. It is one of the most complete specifications of what constitutes a SEE framework and has been adopted by the STSC as the basis for evaluating SEE frameworks. Refer to [NIST 93] for complete definition of the RM.

##### **3.1.2 RM for a Project Support Environment (PSE).**

This RM was produced by the Project Support Environment Standards Working Group (PSESWG, pronounced "peace-wig") of the U. S. Navy's Next Generation Computer Resources (NGCR) program. The RM specifies the services and interfaces that constitute a PSE; it is not a general RM for SEEs. The PSE RM can be used as a guide for understanding PSEs and developing requirements for a specific PSE. It incorporates the NIST/ECMA Frameworks RM for its framework services. Refer to [NGCR 93b] for complete definition of the RM.

#### **3.2 Standards**

##### **3.2.1 General Information**

Standards provide much of the nomenclature encountered in SEE technology and are important within the DoD when it comes to the acquisition and implementation of a SEE. Standards are necessary to implement an open SEE. Without them, the implementor of an open SEE faces an arduous task installing tools and trying to provide interoperability

and commonality among the tools. With standards, "plug and play" tool installation and interoperability are facilitated for the implementor, and common "look and feel" are facilitated for the end-user. Standards can also promote portability of tools from one framework to another and applications from one platform to another.

Standards are specifications that have been widely accepted by a community of users. Standards that are relevant to SEEs have been, and are being, produced by numerous national and international organizations. Three widely known standards organizations, which are involved with SEE technology, are the Institute of Electrical and Electronics Engineers (IEEE), the American National Standards Institute (ANSI), and the International Standards Organization (ISO). Another source for standards, besides standards organizations, is SEE-relevant commercial products that become what are known as *de facto* standards, such as UNIX and Microsoft's Windows. De facto standards often prove to be more popular than *official* standards because they are usually associated with a widely used, successful product, which makes them commercially viable. Official standards, on the other hand, are usually lacking a commercial incentive, may be slow reaching a consensus, and may represent a compromise among conflicting approaches.

The principal SEE-related standards are classified into seven categories:

- framework standards
- graphical user interface (GUI, pronounced "gooey") standards
- data repository standards
- platform or operating system interface standards
- tool interchange standards
- security standards
- other.

Refer to Appendix G for information on specific standards and organizations that develop and promote standards.

### **3.2.2 Conclusions**

The products that come out of SEE technology are affected by two kinds of standards: (1) those standards specifically related to SEEs, and (2) those that have a much broader application. The popularity of the broader standards, such as X-Windows, Ethernet, and Portable Operating System Interface (POSIX), can be an important selling point for SEE

products. In time, today's standards can be supplemented, or even replaced, as new standards become accepted and popular.

Two SEE framework standards, the Portable Common Tool Environment (PCTE) and A Tools Integration Standard (ATIS), will continue to be important and will evolve towards a more complete standard, possibly merging. The Common Ada Programming Support Environment (APSE) Interface Set (CAIS-A) standard will be superseded by a newer, more inclusive standard incorporating work from other standards, possibly adopting the descendant of the PCTE and ATIS. The Portable Common Interface Set (PCIS) activity is tasked with establishing a more inclusive standard incorporating earlier standards work; the current direction of PCIS activity is to define new layers of SEE services on top of the PCTE in which the ATIS will be a major influence. As a result, PCIS may effect the eventual replacement of the PCTE and ATIS (and possibly other standards).

The commercial market influences SEE standards; consequently, the DoD is faced with the dilemma of continuing to emphasize its own standards or adopting more products based on commercial standards. Commercial products provide earlier access to advances in SEE technology and provide interoperability with a greater variety of other products. Limiting the DoD's use of SEE products to those that comply with its standards may actually hamper its goal of flexibility through software interoperability. The DoD policy on commercial-off-the-shelf (COTS) software will probably lead to its adoption of more commercial standards in the future [DRUYUN 93].

(This page is intentionally blank.)

## **4 State of SEE Technology**

In this section are observations concerning the current state of SEE technology and anticipated future directions.

### **4.1 Summary of State-of-the-Practice**

SEEs can be very complex; consequently, SEE frameworks and integrated tools can take a long time to develop. The field is still young, and perceptions of what are appropriate products are growing and changing. Several commercial SEEs have been around for a few years and have been extensively tested by use on multiple projects. SEEs that are available and have some expectation of long-term support vary widely in the features they address and how the features are implemented. In short, SEE technology is still showing some immaturity, but is continuing to evolve.

New SEE framework products, integrated tools and toolsets are continuing to appear on the market. Many have distinctly new approaches; some are implementations of current standards, and some are expansions of CASE tools now addressing the broader application of a SEE. Products and tools that are appearing in the market come from companies with established reputations in Software Engineering technology, from startup companies specializing in SEE technology and from alliances of both kinds of companies. Note that there is still a degree of volatility in the current SEE marketplace; some companies that were believed to be "established" in the marketplace have faced cutbacks in sales, staff and commitments, and may be in danger of going out of business.

Major players for currently available, viable SEE technology products are:

- Atherton Technology
- Cadre Technologies Inc.
- Cap Gemini
- Computer Resources International A/S
- Digital Equipment Corporation
- Electronic Data Systems Corporation
- GIE Emeraude
- Hewlett-Packard Company

- IBM Corporation
- Interactive Development Environments
- Logicon
- Microsoft Corporation
- Rational
- Société Française de Génie Logiciel
- Softlab Inc.
- Sun Microsystems/SunSoft
- Texas Instruments.

#### **4.2 Summary of the State-of-the-Art**

The need to standardize SEE framework interfaces has been recognized with the development of standards like the Portable Common Tool Environment (PCTE); the Common Ada Programming Support Environment (APSE) Interface Set, Revision A (CAIS-A); and the Common Object Request Broker Architecture (CORBA). However, these standards do not go far enough towards developing SEEs. PCTE and CAIS-A are focused to a large extent on a common data repository (repository-centered) as the means to integrate tools into a consistent SEE framework, whereas SEEs based on CORBA are messaging-centered. Issues that must be addressed (since they are issues that address the viability of the SEE) are communication among SEE components, support for defined life cycle development processes, user interface concerns, and administration of the SEE framework. There is no industry-wide consensus on these issues.

#### **4.3 Government Initiatives**

The Federal government has taken an active role in promoting the development of SEE technology. The agencies that have been the most active are the DoD and NIST. Government activities have taken the form of funding organizations to study, evaluate, and facilitate SEE technology, including the funding of the direct development of SEE technology. A partial list of government activities is given here.

#### **4.3.1 The Process-Oriented Software Life Cycle Support Environment (ProSLCSE)**

The ProSLCSE is based on the Air Force's Rome Laboratory's Software Life Cycle Support Environment (SLCSE) and is currently being developed by International Software Systems, Inc. It is intended to be a total life cycle SEE that supports DoD Standard 2167A [ISSI 93].

#### **4.3.2 Software Technology for Adaptable Reliable Systems (STARS) Program**

STARS is a program of the Advanced Research Projects Agency (ARPA). One of the goals of the STARS program is to promote and facilitate megaprogramming.

Megaprogramming is an approach to modular software development and support that emphasizes the reuse of large segments of software. The STARS program works with the Software Engineering Institute (SEI), the Corporate Information Management (CIM) Program, standards organizations, and other groups to evolve open architecture standards and practices to support megaprogramming. The program also works with the vendor community to demonstrate the viability of adaptable, commercially available SEEs.

Three prime contractors (Boeing, Unisys, and IBM) are currently funded by the STARS program to demonstrate SEEs that support megaprogramming. Operational SEEs leading to commercial products are planned for 1994 or 1995.

#### **4.3.3 Integrated Computer-Aided Software Engineering (I-CASE)**

I-CASE is an effort by the Air Force, on behalf of the DoD, to develop a CASE environment and tools to support administrative software development. The effort includes an open, integrated CASE repository, software tools, run-time software components, and user support services. Twenty or more pilot projects at almost as many sites are scheduled to last up to one year and will test the tools and environment at different stages of the software development life cycle. Initial contract award was made in November 1993, but the contractor was subsequently disqualified. The new contractor selection was announced at the 1994 Software Technology Conference.

#### **4.3.4 The Software Technology Support Center (STSC)**

The goals and objectives of the STSC focus on the collection and dissemination within the Air Force of useful information on the role of technology in improving software development and support. This information includes identification and evaluation of products and methodologies and guidance on how to put the technology into practice. The STSC reaches audiences across the entire DoD community through its publications,

such as *CrossTalk* and technology reports, and through its annual Software Technology Conference.

#### **4.3.5 The Integrated SEE Special Interest Group (ISEE SIG)**

The ISEE SIG was created under the umbrella of the NIST's Open Systems Environment Implementors' Workshop (OIW) and was tasked to identify and pull together combinations of standards that can be used to build SEEs. These combinations are referred to as *profiles*. The ISEE SIG replaces the former NIST ISEE Working Group, which was dissolved in 1993.

#### **4.3.6 The Software Engineering Institute (SEI)**

The SEI is chartered to:

- accelerate to common practice, modern software engineering methods, techniques and practices;
- facilitate the adoption and institutionalization of software engineering technology within the DoD community;
- define goals of excellence for the practice of software engineering.

The SEI's primary goal is the transfer of technology. The SEI provides inputs into the STARS program and to NIST as well as pursuing studies in SEE technology.

#### **4.3.7 The Project Support Environment Standards Working Group (PSESWG)**

The funding for the Navy's Next Generation Computer Resources (NGCR) PSESWG (pronounced "peace-wig") effort ended in FY93. The objective of the PSESWG was to select a collection of standards for interfaces to be used in the acquisition of Project Support Environments (PSEs), specifically PSEs that would support the development of mission critical computer resources (MCCR) software. The resulting military standard was to be usable in the procurement of Navy software systems in 1998. Much of the work of the PSESWG has been picked by the SEI and NIST.

#### **4.3.8 Corporate Information Management (CIM) Program**

CIM is a DoD program managed by the Defense Information Systems Agency (DISA). The goal of the program is to apply Information Resource Management (IRM) principles across the entire DoD community. Objectives of the program are:

- elimination of redundant DoD systems
- reduction of maintenance costs by streamlining systems

- establishment of a solid foundation for efficient development of new, effective corporate information systems at reduced costs.

#### **4.4 Future Directions and Trends**

Major improvements in SEE technology will require time to address and resolve issues such as the following:

- Vendors are recognizing that they alone cannot supply a full set of software development tools for the entire software life cycle; the cost of developing and maintaining the tools is too high. This problem has led to the formation of vendor groups that are formed to supply the full set of tools. The future may see a few large vendors becoming sole sources for the full set.
- Standards bodies may be taking too much time to reach consensus and closure on standards; SEE builders, rather than waiting, are creating de facto standards. Standards bodies need to be willing to drop obsolete standards in favor of de facto standards when the de facto standards are clearly superior.
- Programs such as the NIST ISEE SIG require time to further their work in creating profiles of standards.
- Communication between academia, vendors, and users must be promoted so that important user needs are correctly perceived and implemented.

Areas that have been insufficiently addressed up to now by SEE technology and from which improvements are to be expected are:

- software development process modeling
- software development process definition and capture
- software development process enactment and enforcement
- reuse of existing software and software project artifacts
- on-going support for existing software (legacy software)
- human skills and training
- security.

##### **4.4.1 Software development process**

Future SEE technology will include increased emphasis on modeling the software development process—allowing for more detailed definitions of the user roles, tasks, and

rules that define the organization's required process. It follows that future SEE technology will include increased emphasis on process enactment.

#### **4.4.2 Reuse**

Support for reuse will involve storing representations of work products from all the software life cycle phases. There is also potential for great improvements in reuse through support for the analysis of application domains with a view toward reuse.

#### **4.4.3 Security**

Future SEE technology will have more support for security features, which will reside within the SEE framework. Security features will support traditional security requirements, such as access and permissions, while support for multilevel security will mature from the efforts of the DoD and National Security Agency.

## **5 Applications**

This section addresses the use of SEE technology in an organization—why, when, and how. Technical hints and acquisition are also covered in this section.

### **5.1 Why Do You Use SEE Technology?**

An organization adopts SEE technology to improve its software quality and software development productivity. The improvements are achieved by automating many of the labor-intensive and often trivial tasks of software development, by reducing the generation of human errors in the software development process, and by reusing the work of others. The need for SEE technology increases with growth in the size of the software development teams and in the complexity of the software.

Adoption of SEE technology can provide some or all of the following capabilities:

- Assistance in defining the software process
- Support for widespread and consistent use of process automation
- Support for a well-defined software process
- Data sharing among tools
- Intertool messaging
- Common user interface to multiple (CASE) tools
- Integrated life cycle processes, e.g., configuration management
- Improved and consistent group coordination and communication
- Automated collection of metrics.

### **5.2 When Do You Use SEE Technology?**

The selection and implementation of a particular SEE technology presuppose a commitment to a software development methodology, or methodologies. The software development methodology emerges from an organization's implementation of a defined software development process [FOWLER 90]. With the establishment of a defined and managed process, the software professionals in the organization are better able to evaluate, select, and adopt a SEE [HUMPHREY 90].

How does an organization determine if it is ready to use SEE technology? A SEE will be of particular interest to an organization that has adopted advanced methods for the design, management, and maintenance of large or complex software projects. Acquiring a SEE should not be viewed as a solution to out-of-control software development, but rather as an aid to achieving a well-defined process for managing software development in the organization. The following list, though not intended to be comprehensive, provides some essential attributes of an organization implementing a SEE:

- Experience in the use of CASE tools
- Funding allocated for equipment, software, training, and the additional roles required for managing and supporting the SEE
- High degree of receptiveness to change
- Opportunities for widespread use of the services provided by the SEE
- Management commitment to provide people and money resources
- People within the organization with expertise in serving as champions of change
- Little or no schedule constraint for an initial project.

The purpose of a SEE is to support and improve Software Engineering within an organization. Improvement is measured in terms of the reduction in effort and total life cycle cost to produce, support, and maintain a software application. In particular, the purpose of the SEE is to automate those aspects of Software Engineering that can improve the quality of the software being produced and raise the productivity of the software developers.

### **5.3 How Do You Use SEE Technology?**

Reaping the most benefits from SEE technology comes from a well-planned implementation strategy and careful selection of the SEE. The insertion of SEE technology starts with having a defined process in place, understanding the process, and then developing plans and goals to improve the process supported by SEE technology. *IEEE Std 109-1992: IEEE Recommended Practice for the Evaluation and Selection of CASE Tools* offers guidelines and recommended practices that are adaptable to the selection of a SEE.

## **5.4 Important Issues**

SEE products vary considerably in how they can be adapted to the requirements of specific projects and organizations. Selection of a SEE product should be based on how it can accommodate the particular approach an organization uses to develop software.

Often an organization has one core process model that can be tailored or customized for all the projects that the organization is likely to handle. It may need to be tailored to the process for innovative research on new software products, or for maintenance of existing products, or for production of well-defined, mainstream products. The process may even need to be different for applications with special features such as real-time control, database-intensive programs, or high-resolution graphics.

A SEE product should be selected to match the generic process model, but also to have the flexibility to customize to specific projects. Customization should allow for variations in programming languages, design methodologies, documentation standards, and in software development life cycle models (rapid prototyping, waterfall, spiral, etc.). Although some SEE products are oriented to a specific process, design methodology, documentation standard, or programming language, most are flexible enough to adapt to the needs of different projects.

In the future, many SEEs will support a software life cycle process model precisely defined by the user (within some constraints). The SEE will also help the user to evolve and optimize that process and to manage the impacts inherent in change. A major challenge will be in balancing the control necessary for a stable environment against the flexibility necessary for adapting and improving the process.

## **5.5 Management and Business Matters**

Adoption of SEE technology may expose an organization to risks in many areas:

- The user may expect too much from the technology without sufficient focus on developing strong software engineering and management skills.
- Due to the large investment usually required for SEE technology, the return on investment is low (if not negative) until software process is well-defined.

- The return on investment is also affected by the training time and the time needed to understand how best to use the product.
- With SEE technology being relatively new and evolving, a strong decision to commit to today's technology must be weighed against trends and tomorrow's needs.
- There is a high probability of underestimating the required resources to manage and support the SEE.

Maturity in technology adoption comes in stages: from trying it out on a first project, moving on to official acceptance and enforcement resulting in integration into the "corporate culture." SEEs will be most successful where individuals are trained and experienced in effective Software Engineering principles and where the organization has developed a culture of commitment to software quality.

## **5.6 Technical Hints**

Once an organization has determined that it is ready for a SEE, the key to selecting the right SEE products is matching them to the software development needs of the organization. There is no "best" product to suit everyone's needs, nor to suit even the needs of all Air Force organizations. However, there may be a best SEE product for a given organization's software development process. In the following paragraphs are some questions to seriously consider.

### **5.6.1 Organizational needs**

The primary step in selecting a SEE is to determine the organization's needs. The following questions give a flavor of what should be asked of the organization:

- (1) If the organization supports more than one software development process model, what are the specific support needs for each model?
- (2) What constraints are placed on the organization needs, e.g., integration of CASE tools already in use, specific platform requirements, specific network requirements?
- (3) What are the organization's specific needs with respect to integration, support for software life cycle phases, customization, compliance to standards, specialization?
- (4) What are the organizational needs to effectively develop and maintain software?
- (5) What are the resources the organization can realistically apply to software projects?
- (6) What size of projects does the organization intend to support?
- (7) Does the organization require evolutionary implementation of SEE technology (starting with a small set of capabilities and growing incrementally)?

- (8) What are the training needs?

### **5.6.2 Product requirements**

The following questions give a flavor of determining the organization's requirements for the SEE product:

- (1) Has the SEE product supported projects of comparable size to the organization's needs?
- (2) Does the product's history indicate that it is sound and mature?
- (3) Has the product been applied in a similar application domain?
- (4) Are evaluations of the product available?
- (5) What are the recommended training requirements?
- (6) Is the customer given full rights and access to source code, and if so, at what price?
- (7) How frequent are releases of the product?
- (8) What consideration is given to maintaining upward compatibility from one release of the product to another?
- (9) What are the total maintenance costs and does the maintenance agreement include future releases of the product?
- (10) Does a user group exist for discussing problems, enhancements, etc.?
- (11) Does the product fit with other products that are being considered?

### **5.6.3 Vendor evaluation**

The following questions give a flavor of evaluating the vendor of the SEE product:

- (1) Will the vendor provide an evaluation copy of the SEE product for a specified period of time?
- (2) Will the vendor support the organization's evolutionary implementation of SEE technology (starting with a small set of capabilities and growing incrementally), if that is the organization's strategy?
- (3) Is there a vendor staff dedicated to user support?
- (4) Does the vendor live up to commitments and promises?
- (5) Are future projections for the vendor positive and stable?
- (6) Does the vendor provide a responsive, helpful hot-line service?
- (7) Does the vendor provide installation and consultation support?

## **5.7 Acquisition of a SEE**

It is recommended that SEE products be selected with the help of qualified technical advisors who have current information and experience in SEE technology. One of the main roles of the STSC is to help organizations identify and acquire appropriate, advanced software technology.

## **6 SEE TECHNOLOGY PRODUCTS**

This section discusses the process used by the STSC SEE domain team to gather information on available SEE technology products. The specific information that was gathered is contained in the appendices of this report.

### **6.1 Scope**

The SEE technology products that appear in this report were selected as being identified as a SEE framework or a populated SEE framework. A SEE framework is a procurable product that facilitates and supports tools integration and interoperability and data sharing by a repository. A populated SEE framework is a SEE framework that has integrated into it end-user tools that support one or more phases of the software life cycle.

### **6.2 The Information Gathering Process**

The information gathering process used for this report consisted of the following steps:

- (1) Survey the SEE technology products market.
- (2) Identify the products that meet the criteria outlined in paragraph 6.1.
- (3) Obtain complete descriptions of the products from the vendors through use of a STSC questionnaire and by interview, if possible.
- (4) Identify users of the selected products.
- (5) Obtain product critiques from the users through use of a STSC survey sheet and by follow-on interviews, wherever possible.

### **6.3 Product Lists**

A list of the selected SEE technology products, sorted by source (vendor), appears in Appendix C.

#### **6.4 Product Sheets/Vendor Questionnaires**

Vendor questionnaires were sent to each of the vendors on the product list. The returned questionnaires were summarized onto product sheets and the features matrix. The product sheets appear in Appendix D and the features matrix in Appendix F.

#### **6.5 Product Critiques**

The users' experiences with the selected SEE technology products were captured on product critique forms, which had been sent to the users. On the form, the user was asked not only about experience with the product but also about software experience in general. Even though the users' names were provided by the vendors, the product critiques were returned directly to the STSC. Critiques were not returned on every product. The completed critique forms are in Appendix E.

#### **6.6 Criteria**

The need for adequate criteria to evaluate SEE technology products became apparent during the information gathering process. The STSC decided to take an active role within the DoD to work at improving criteria for SEE technology product evaluations. To do so, the STSC established the informal DoD SEE Criteria Working Group (SEECRIT WG), which has started pulling together criteria from interested agencies within the Federal government. Going beyond information gathering, the goal of the SEECRIT WG is to provide adequate criteria for the use of organizations that are acquiring SEE technology.

## **7 Conclusions**

SEE technology can be a major building block in the software development improvement process. By introducing automated software engineering into software development and post-development support, long-term improvements in productivity and quality can be expected, although significant costs should be expected in the short-term.

SEE technology is not a "silver bullet"—it is not a replacement for a good software development team nor for a well-defined software development and support process. In an organizational culture that is structured, managed, and understands its processes, a SEE can enhance productivity and quality. Insertion of SEE technology should be a facet of a multifaceted approach to software development and support improvement.

(This page is intentionally blank.)

**Appendix A: Acronyms and Terms**

AFB	Air Force Base
AJPO	Ada Joint Program Office
ANSI	American National Standards Institute
APSE	Ada programming support environment
ARPA	Advanced Research Projects Agency
ATIS	A Tools Integration Standard
CAD/CAM	computer-aided design/computer-aided manufacturing
CAIS-A	Common APSE Interface Set, Revision A
CASE	Computer-Aided Software Engineering
CAST	Computer-Aided Software Testing
CDE	Common Desktop Environment
CDIF	CASE Data Interchange Format
CIM	Corporate Information Management
CIS	CASE Integration Services
CM	configuration management
CMM	Capability Maturity Model
CORBA	Common Object Request Broker Architecture
COSE	Common Open Software Environment
COTS	commercial off-the-shelf
CRI	Computer Resources International
DBMS	database management system
DDL	data definition language
DEC	Digital Equipment Corporation
DISA	Defense Information Systems Agency
DML	data manipulation language
DoD	Department of Defense
DOD-STD	DoD Standard
ECMA	European Computer Manufacturers Association
EDS	Electronic Data Systems (Corporation)
EIA	Electrical Industries Association
e-mail	electronic mail
FFRDC	Federally Funded Research and Development Center
GUI	graphical user interface
IBM	International Business Machines
I-CASE	Integrated CASE
IDE	Interactive Development Environments
IEEE	Institute of Electrical and Electronics Engineers
IGES	Initial Graphics Exchange Specification
IPSE	integrated project support environment
IRDS	Information Resource Dictionary System
IRM	Information Resource Management
ISEE	Integrated Software Engineering Environments
ISO	International Standards Organization

*Appendix A: Acronyms and Terms*

ISSI	International Software Systems, Inc.
MCCR	mission critical computer resources
NAPI	North American PCTE Initiative
NGCR	Next Generation Computer Resources
NIST	National Institute of Standards and Technology
NSA	National Security Agency
OIW	Open Systems Environment Implementors' Workshop
OMA	Object Management Architecture
OMG	Object Management Group
O-O	Object-Oriented
OO-ALC	Ogden Air Logistics Center
ORB	Object Request Broker
OSF	Open Software Foundation
PC	Personal Computer
PCIS	Portable Common Interface Set
PCTE	Portable Common Tool Environment
PDES	Product Data Exchange Specification
POSIX	Portable Operating System Interface
ProSLCSE	Process-oriented Software Life Cycle Support Environment
PSE	Project Support Environment
PSESWG	Project Support Environment Standards Working Group
PTI	public tool interface
RM	reference model
SCUD	(Name of a Soviet-built medium range missile used by Iraq in the Gulf War)
SEE	software engineering environment
SEECRIT WG	SEE Criteria Working Group
SEI	Software Engineering Institute
SEPG	Software Engineering Process Group
SFGL	Société Française de Génie Logiciel
SIG	Special Interest Group
SLCSE	Software Life Cycle Support Environment
SLOC	source lines of code
SQL	Structured Query Language
STARS	Software Technology for Adaptable Reliable Systems
STC	Software Technology Conference
STEP	Standard for the Exchange of Product Model Data
STSC	Software Technology Support Center
SVR4	UNIX System V version 4
TCOS	Technical Committee on Operating Systems
UI	UNIX International

(This page is intentionally blank.)

## **Appendix B: Glossary**

The definitions in this glossary are, in many cases, alternatives to the way terms may be defined in the body of the report. Also, terms may be defined in this glossary that do not appear in the report; these terms are SEE-related and are provided for information to the reader.

architecture	(i.e., <i>digital system architecture</i> ) A system or style of building having certain characteristics of structure. For digital computer systems, includes the hardware and software components (structure), their <i>interfaces</i> , and the <i>execution concept</i> that underlies system processing [HOROWITZ 91].
assessing a SEE	Describing accurately with reference to an accepted model what capabilities of a SEE are to support and integrate a range of CASE tools and to support specific software development activities [NIST 93a]. (For comparison, refer to <i>evaluating a SEE</i> .)
CASE	Computer-aided software engineering—the use of computer-based support in the software development process [SEI 93].
CASE methodology	An "automatable" structured methodology that defines a disciplined, engineering-like approach for all or some aspects of the development and maintenance of software [MCCLURE 89].
CASE technology	A software technology that provides an automated, engineering discipline for software development, maintenance, and project management; includes automated structured methodologies and automated tools [MCCLURE 89].
CASE tool	A specific computer-based product aimed at supporting one or more software engineering activities within a software development process [SEI 93].
certification	The authoritative endorsement of a software product's suitability for use in specific applications [STSC 92].
control integration services	Framework integration services that allow the passing of messages and information from one tool to another in order to provide the coordination and communication of tool functions [ROME 93].

Appendix B: Glossary

data integration services	Framework integration services that allow a tool to deposit, retrieve, and share its data with other tools via a common life cycle information repository [ROME 93].
de facto	(1) In reality, (2) the actual fact.
enforcement	(Re: process assurance) The activities used to ensure that process enactment conforms to process constraints [FEILER 92].
environment	The combination of all hardware, software, people, and procedures involved in a design activity [NIST 91].
evaluating a SEE	Determining how well the SEE meets a set of customer requirements [NIST 93a]. (For comparison, refer to <i>assessing a SEE</i> .)
framework	A set of commonly needed facilities, key integration components, and support for higher level constructs than those found in typical operating systems [NIST 93]. (See <i>SEE Frameworks</i> .)
horizontal integration	Tools are integrated in a SEE across the multiple phases of the software life cycle [CUTHILL 93].
lower-CASE	Computer-Aided Software Testing (CAST) [STSC 93].
maturity level	A well-defined evolutionary plateau toward achieving a mature software process [PAULK 93].
model	A representation used to visualize a concept.
object-oriented	A systems development method where the basic unit of design is the object [WHITE 93].
presentation integration services	Framework integration services that provide a common "look-and-feel" for tools integrated into an environment [ROME 93].
process definition	Defining the elements necessary to accomplish an end goal. Among these elements are tasks, artifacts to be consumed (inputs) and produced (outputs) by tasks, resource types necessary to perform tasks, e.g., personnel roles, machine types, tool types, and the dependencies between tasks [ROME 93].
process enactment	The execution of a project in compliance with a process definition [ROME 93].

process enforcement	A part of process enactment that does not allow personnel to deviate from the defined process [ROME 93].
process instantiation	Making a process definition enactable by appropriately assigning available organizational resources, e.g., actual personnel, machines, to tasks and establishing the mechanisms necessary to support process enactment [ROME 93].
process integration services	Framework integration services that support the definition, installation, enactment, analysis, and abstraction of software engineering processes [ROME 93].
quality software	Software that meets the users' requirements or expectations.
reference model	A conceptual and functional framework that helps experts to describe and compare systems [BROWN 92].
repository	A mechanism for storing and organizing all information concerning a software system [MCCLURE 89].
SEE	(See <i>software engineering environment</i> .)
SEE Frameworks	A set of (relatively) fixed infrastructure capabilities that provide support for processes, objects, or user interfaces [NIST 93]. (For comparison, see <i>tools</i> .)
software capability evaluation	An appraisal by a trained team of software professionals to identify contractors who are qualified to perform the software work or to monitor the state of the software process used on an existing software effort [PAULK 93].
Software Engineering	The planned process of producing well-structured, reliable, good-quality, maintainable software systems within reasonable time frames [NIST 93].
software engineering environment	The system that provides automated support of the engineering of software systems and the management of the software process [NIST 93].

*Appendix B: Glossary*

software process	A set of activities, methods, practices, and transformations used to develop and maintain software and the associated products [PAULK 93].
software process assessment	An appraisal by a trained team of software professionals to determine the state of an organization's current software process, to determine the high-priority software process-related issues facing an organization, and to obtain the organizational support for software process improvement [PAULK 93].
software process capability	Describes the range of expected results that can be achieved by following a software process [PAULK 93].
software process maturity	The extent to which a specific process is explicitly defined, managed, measured, controlled, and effective [PAULK 93].
software process performance	Represents the actual results achieved by following a software process [PAULK 93].
tool	A piece of software that is not part of the SEE framework and which calls upon the services provided by the SEE framework [BROWN 92].
upper CASE	CASE tools aimed at the requirements analysis and high level design portions of the software life cycle [STSC 92a].
validation	The process of confirming that the software, i.e., documentation and computer program, satisfies all user requirements when operating in the user environment [STSC 92].
verification	The process of confirming that the products of each software development phase, e.g., requirements definition, design, and coding, are complete, correct, and consistent with reference to the products of the preceding phase [STSC 92].

(This page is intentionally blank.)

## **Appendix C: SEE Products List by Source**

*Software Technology Support Center*

Source	SEE Product (Version)	Address	Contacts
Alslys, Inc.	FreedomWorks (1.3)	10251 Vista Sorrento Parkway San Diego, CA 92121	M: Bill Hart 619-457-2700 T: 800-995-2579
Atherton Technology	Software Backplane (3.0.2)	39350 Civic Center Drive Suite 430 Fremont, CA 94538	M: Jim Conlon 510-494-8411 T: Chan Drawdy 510-494-8411
AT&T	Metatool Specification Driver (2.0)	10 Independence Blvd. Warren, NJ 07059	M: Ed Cartier 908-580-5719 T: Kristin Kassas 508-960-6473
Bull HN Information Systems Inc.	Integrated System Development (1.0)	300 Concord Road Billerica, MA 01821	M: Steve Barry 508-294-4572 T: Cameron McLean 508-294-4623
Cap Gemini Innovations	Process Weaver (2.0)	Avenue of the Americas 29th Floor New York, NY 10036	M: Larry Proctor 915-947-7183 T: Vincent Jorrano 800-400-1511
Centerline Software	CodeCenter/ObjectCenter (4.0)	7926 Jones Branch Drive, Suite 1000 McLean, VA 22102	M: David Lyddane 703-749-1100 T: Norman Aroesty 703-749-1100
CRI Inc.	LIFE*CYCLE (2.2)	3245 146th Place, SE #270 Bellevue, WA 98007	M: John Wilkerson 206-643-7443 T: Andy Bliven 206-643-7443
Digital Equipment Corp.	COHESION ASD/SEE (2.3)  COHESION Team/SEE (1.0)  COHESIONworX (2.0)	4 Results Way, MR04-3/M6 Marlboro, MA 01752  4 Results Way, MRO4-3/H6 Marlboro, MA 01752	M: Andrew Willinger 508-467-4041 T: Rob Straight 603-884-2578 M: Paul Hambelton 508-467-6968 T:  M: Andrew Willinger 508-467-4041 T:
GIE Emeraude (Mark V Systems, USA distributor)	Emeraude V12 (12.4)		M:  T:

M: Marketing Contact

T: Technical Contact

*Appendix C: SEE Products List by Source*

Source	SEE Product (Version)	Address	Contacts
Hewlett-Packard Company	SoftBench, SoftBench/C++ (3.3)	3404 East Harmony Road Fort Collins, CO 80524	M: Chuck Dahl 206-643-8784 T: Michale Barrat 206-644-3253
IBM Federal Systems Co.	AIX SDE Work Bench/6000, AIX SDE Integrator/6000	6710 Rockledge Drive Bethesda, MD 20817	M: Barbara Jenness 301-803-6235 T: Marvin Harrison 416-448-6231
IDE	C - Development Environment Software Through Pictures (1.2) OOSD/Ada - Software Through Pictures (1.2) STRAND-RT - Software Through Pictures (4.2D)	595 Market Street 10th Floor San Francisco, CA 94105	M: Jeff Morris 415-543-0900 T: 800-444-7871 M: Chris Lalli 415-543-0900 T: 800-444-7871
International Software Systems, Inc.	ProSLCSE (2.2) [?]	5430 Research Blvd. Echelon IV, Suite 250 Austin, TX 78759	M: Priscilla Yeh 512-338-5721 T:
IPSYS Software	IPSYS ToolBuilder (1.3)	28 Green Street Newbury, MA 01951	M: Donna MacDonald 508-463-0006 T: 508-463-0006
Manager Software Products, Inc.	METHODMANAGER (2.7.0)	131 Hartwell Avenue Lexington, MA 02173	M: Carl Olofson 617-863-5800 T: Carl Olofson
Mark V Systems	EAST Environment ( ) ObjectMaker (3.0)	16400 Ventura Blvd., #303 Encino, CA 91436	M: Mo Bjornestad 818-995-7671 T: 818-995-7671
McDonnell Douglas Information Systems Ltd.	Pro-IV Workbench (1.0)	325 J.S. McDonnell Blvd. Mailcode 3065600 Hazelwood, MO 63042	M: Al Weiss 314-233-6068 T: 800-225-7760
Mesa Systems Guild, Inc.	Cradle SEE (1.8)	168 Ninth Street Providence, RI 02906-2931	M: Michael R. Bousquet 401-421-9390 T: Brain Brady 401-421-9390
Objectime Limited	Objectime (4.1.1)	340 March Road, Suite 200 Kanata, Ontario, Canada K2K 2E4	M: Garth Gullekson 800-567-TIME T: 613-591-3400

M: Marketing Contact

T: Technical Contact

*Software Technology Support Center*

Source	SEE Product (Version)	Address	Contacts
Objective Systems	Objectory (3.3.1)	4 Greenwich Office Park Greenwich, CT 06831	M: Per Kroll 203-625-7250 T: Per Kroll
Rational	Apex (1.2)	3320 Scott Blvd. Santa Clara, CA 95051-0951	M: Jerry Rudisin 408-496-3600 T: Kent Mitchell 408-496-3600
Software Systems Design, Inc.	AISLE Tools (2.0)	3627 Padua Avenue Claremont, CA 91711	M: Dr. Tom Radi 909-625-6147 T: Dr. Tom Radi
Texas Instruments	Information Engineering Facility (5.1)	1400 S. Potomac Street, Suite 101 Aurora, CO 80012	M: Betty Iura 303-369-1329 T: Barbie Stafford 303-369-1327
Virtual Software Factory Limited	VSF, MWB-SF/AWB-SF (3.8)	13873 Park Center Road, Suite 218 Herndon, VA 22071	M: Steve Shap 703-318-1180 T: Charles Williams 703-318-1180
Vista Technologies, Inc.	PCTE Workbench (1.1)	1100 Woodfield Rd, Suite 437 Schaumburg, IL 60173-5124	M: Edith Ludwig 708-706-9300 T: David Hurst 708-706-9300

M: Marketing Contact

T: Technical Contact

## **Appendix D: SEE Product Sheets**

## INDEX OF PRODUCTS

AISLE Tools .....	D-3
AIX SDE Integrator/6000 .....	D-4
AIX SDE Work Bench/6000 .....	D-4
Apex .....	D-5
C - Development Environment Software Through Pictures .....	D-6
CodeCenter/ObjectCenter .....	D-7
COHESION ASD/SEE .....	D-8
COHESION Team/SEE .....	D-9
COHESIONworX .....	D-10
Cradle SEE .....	D-11
EAST Environment .....	D-12
Emeraude V12 .....	D-13
FreedomWorks .....	D-14
Information Engineering Facility .....	D-15
Integrated System Development .....	D-16
IPSYS ToolBuilder .....	D-17
LIFE*CYCLE .....	D-18
Metatool Specification Driver .....	D-19
METHODMANAGER .....	D-20
Objectime .....	D-21
ObjectMaker .....	D-22
Objectory .....	D-23
OOSD/Ada - Software Through Pictures .....	D-24
PCTE Workbench .....	D-25
Process Weaver .....	D-26
Pro-IV Workbench .....	D-27
ProSLCSE .....	D-28
SoftBench .....	D-29
SoftBench/C++ .....	D-29
Software Backplane .....	D-30
STRAND-RT - Software Through Pictures .....	D-31
VSF, MWB-SF/AWB-SF .....	D-32

## AISLE Tools

**Product Information:**

Version Number: 2.0  
Date of Latest Release: June 1993  
Date of First Release: Sept 1989  
Frequency of Updates: 2-3 per year  
Number Sold: 200+

**Pricing:**

Single User Price: From \$5000 up  
Site License: Contact vendor.  
Multicopy Price: Contact vendor.

GSA Price:  
Maintenance Price: 15 percent per year.

**Vendor Contact Information:**

Software Systems Design, Inc.  
Address: 3627 Padua Avenue  
Claremont, CA 91711

In Business Since: 1986

**Marketing Contact:** Dr. Tom Radi

Phone Number: 909-625-6147

Fax Number: 909-626-9667

E-mail Address:

**Technical Contact:** Dr. Tom Radi

Phone Number: 909-625-6147

Fax Number: 909-626-9667

E-mail Address:

Bulletin Board System:

Customer Support:

**Platform(s)/Operating System(s):**

VAX/VMS, Unix, Sun, HP, Silicon Graphics 88 Open, 80386/DOS, 80386/Unix.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**  
4 MB RAM.

**Description/Purpose:**

Ada Integrated Software Lifecycle Environment (AISLE) is an integrated set of software tools that assist in the design, development, documentation, and testing of Ada computer programs. The tools family is centered around Ada Design Language (ADADL), an Ada-based Program Design Language (PDL). ADADL fully satisfies DoD Directive 3405.2, which requires the use of an Ada PDL in the development of new programs for the DoD. The latest addition to AISLE is QualGen, a tool that helps improve Ada quality.

**AIX Software Development Environment (SDE) Work Bench/6000**  
**AIX SDE Integrator/6000**

**Product Information:**

Version Number: 1.2  
Date of Latest Release: Sept 1992  
Date of First Release: May 1992  
Frequency of Updates: Semiannual  
Number Sold: Available on request.

**Pricing:**

Single User Price: Workbench \$1,300 - \$10,410  
Integrator \$1,825 - \$14,610  
Site License: No  
Multicopy Price: Workbench 10 percent  
discount per user.  
Integrator 10 percent  
discount per user.  
GSA Price: Yes  
Maintenance Price: None

**Vendor Contact Information:**

IBM Federal Systems Co. Inf. Center  
Address: 6710 Rockledge Drive  
Bethesda, MD 20817

In Business Since: 1924

**Marketing Contact:** Barbara Jenness

Phone Number: 301-803-6235

Fax Number: 301-803-6552

E-mail Address:

**Technical Contact:** Marvin Harrison

Phone Number: 416-448-6231

Fax Number: 416-448-4414

E-mail Address: harrison@vnet.ibm.com

Bulletin Board System: None

Customer Support: IBM Support Center

**Platform(s)/Operating System(s):**

IBM AIX Version 3 Release 2 for RISC System/6000 (5756-030).

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

16 MB RAM minimum (32 MB recommended); 12 MB disk space for Software Development Environment (SDE) Workbench/6000, 2 MB disk space for SDE Integrator/6000.

**Description/Purpose:**

SDE Workbench/6000 provides the basis for an integrated software development environment. SDE Workbench/6000's visual and control integration services are based on Hewlett-Packard's SoftBench integration framework technology. These services allow users to create a customized, integrated software development environment. SDE WorkBench/6000 includes a set of core development tools for software development.

SDE Integrator/6000 is a companion product that enables users to integrate new and existing tools such that they can utilize the SDE Workbench/6000's visual and control services and be integrated with other tools. SDE Integrator/6000 also enables users to link tools together to form and execute a basic process.

## Apex

**Product Information:**

Version Number: Apex 1.2  
Date of Latest Release: Dec 1993  
Date of First Release: Aug 1993  
Frequency of Updates: 6 - 9 months  
Number Sold: 1,000 licenses

**Pricing:**

Single User Price: \$18,500  
Site License: N/A  
Multicopy Price: N/A

GSA Price: N/A  
Maintenance Price: \$250/month

**Vendor Contact Information:**

Rational  
Address: 3320 Scott Blvd.  
Santa Clara, CA 95051-0951

In Business Since: 1981

**Marketing Contact:** Jerry Rudisin

Phone Number: 408-496-3600  
Fax Number: 408-496-3636  
E-mail Address: gjr@rational.com

**Technical Contact:** Kent Mitchell

Phone Number: 408-496-3600  
Fax Number: 408-496-3636  
E-mail Address: kdm@rational.com

Bulletin Board System:

Customer Support: 800-433-5444

**Platform(s)/Operating System(s):**

Sun Sparc or IBM RS/6000 workstations.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

32 MB RAM.  
1.2 GB Disk.  
Motif User Interface.

**Description/Purpose:**

The Rational Apex is an integrated, interactive, and highly automated software engineering support environment for Ada development projects. The purpose of Rational's Apex is as follows:

- Improve productivity through right tool integration.
- Save time and prevent error during coding.
- Reduce recompilation.
- Solve problems with configuration management.
- Capture and enforce architecture.
- Support prototyping and simulation.
- Work with Ada compilers from all vendors.

## C - Development Environment Software Through Pictures

**Product Information:**

Version Number: 1.2  
Date of Latest Release: 1993  
Date of First Release: 1991  
Frequency of Updates: Quarterly  
Number Sold: 500 +

**Pricing:**

Single User Price: \$11,000  
Site License: Contact vendor.  
Multicopy Price:

GSA Price: Contact vendor.  
Maintenance Price:

**Vendor Contact Information:**

IDE  
Address: 595 Market Street,  
10th Floor  
San Francisco, CA 94105

In Business Since:

**Marketing Contact:** Jeff Morris

Phone Number: 415-543-0900

Fax Number: 415-543-0145

E-mail Address: Morris@ide.com

**Technical Contact:** Tech. Support

Phone Number: 800-444-7871

Fax Number: 415-543-1314

E-mail Address: techstaff@ide.com

Bulletin Board System: Yes

Customer Support: Yes

**Platform(s)/Operating System(s):**

Sun-SPARCstation.  
HP300/400/700/800 - HP-UX.  
DECAlpha.  
IBM RS/6000-AIX.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

16 MB RAM; 50 MB Disk; X11 based Window Managers; Open Windows - Open Look; DECAlpha; Ethernet.

**Description/Purpose:**

C Development Environment automatically creates Structure Charts and Data Structure Diagrams from C source code. Once created, the diagrams populate the data dictionary which allows for automatic documentation generation. Tool Supports incremental code generation and incremental Reverse Engineering to diagrams. Can also be used with Refine FORTRAN to Reverse Engineer FORTRAN code.

### CodeCenter/ObjectCenter

**Product Information:**

Version Number: 4.0  
Date of Latest Release: 1992  
Date of First Release: 1988  
Frequency of Updates: Semiannual  
Number Sold: 17,000

**Pricing:**

Single User Price: \$2,995/\$3,995  
Site License:  
Multicopy Price: 1-7 \$2,396/3,196  
8-15 \$2,248/\$2,996  
16-30 \$3,097/\$2,797  
31-50 \$1,797/\$2,397

GSA Price: Same as above.  
Maintenance Price: \$395/\$520 year.

**Vendor Contact Information:**

Centerline Software  
Address: 7926 Jones Branch Drive, Suite 1000  
McLean, VA 22102

In Business Since: 1988

**Marketing Contact:** David Lyddane

Phone Number: 703-749-1100

Fax Number: 703-749-1108

E-mail Address: jsaucedo@centerline.com

**Technical Contact:** Norman Aroesty

Phone Number: 703-749-1100

Fax Number: 703-749-1108

E-mail Address: aroesty@centerline.com

Bulletin Board System:

Customer Support:

**Platform(s)/Operating System(s):**

SUN - Solaris 1.0, Solaris 2.0.  
DG - DG/UX.  
HP - HP-UX 8.0/9.0.  
DEC - ULTRIX 4.3.  
IBM - AIX 3.X.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

16 MB RAM, 30 MB disk space, 50 MB swap.

**Description/Purpose:**

C and C++ programming environment. Construction, debug, and testing of C and C++ programs.

## COHESION ASD/SEE

**Product Information:**

Version Number: 2.3  
Date of Latest Release: 1 October 1993  
Date of First Release: October 1991  
Frequency of Updates: 6 months  
Number Sold: >1,000 seats

**Pricing:**

Single User Price: \$14,200  
Site License: N/A  
Multicopy Price: Digital volume and corporate discounts apply

GSA Price: Same as std. prices  
Maintenance Price:

**Vendor Contact Information:**

Digital Equipment Corporation  
Address: 4 Results Way, MR04-3/M6  
Marlboro, MA 01752

In Business Since: 1957

**Marketing Contact:** Andrew Willinger

Phone Number: 508-467-4041

Fax Number: 508-467-3106

E-mail Address: willinger@mr4dec.enet.dec

**Technical Contact:** Rob Straight

Phone Number: 603-884-2578

Fax Number: 603-884-3163

E-mail Address:

Bulletin Board System:

Customer Support:

**Platform(s)/Operating System(s):**

VAX/Open VMS.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**  
VAX 3100, 32 MB memory, 1 GB disk, VMS 5.5-2.

**Description/Purpose:** The COHESION AeroSpace Defense (ASD)/SEE is Digital's software engineering environment on VAX/VMS for the aerospace and defense industry. A unique aspect of the COHESION ASD/SEE solution is "one-stop shopping." It brings together products from several vendors to meet the needs of projects in many application areas. It includes a comprehensive set of COTS products customized to the aerospace and defense communities where COTS solutions do not meet their needs. The COHESION ASD/SEE solution supports software development across the entire software development life cycle. It is designed primarily to support programming-in-the-large, which is considered to be a department/program development effort involving several teams and 500,000 lines of code or more. The COHESION ASD/SEE solution is also useful for programming-in-the-medium, involving 50,000 to 500,000 lines of code.

## COHESION Team/SEE

**Product Information:**

Version Number: 1.0  
Date of Latest Release: 26 March 1994  
Date of First Release:  
Frequency of Updates:  
Number Sold:

**Pricing:**

Single User Price: \$5500 (per concurrent user)  
Site License:  
Multicopy Price: Digital volume discount  
agreements apply.

GSA Price:  
Maintenance Price:

**Vendor Contact Information:**

Digital Equipment Corporation  
Address: 4 Results Way, MRO4-3/H6  
Marlboro, MA 01752

In Business Since: 1957

**Marketing Contact:** Paul Hambelton

Phone Number: 508-467-6968  
Fax Number: 508-467-3106  
E-mail Address: hambelton@mr4dec.enet.dec

**Technical Contact:**

Phone Number:  
Fax Number:  
E-mail Address:

Bulletin Board System:  
Customer Support:

**Platform(s)/Operating System(s):**

Alpha AXP / DEC OSF/1 AXP  
HP 9000 series / HP-UX  
SPARCstation / SUNOS

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**  
64/96MB memory, 1GB disk, Motif, DCE, TCPIP

**Description/Purpose:**

A comprehensive heterogeneous software engineering environment, COHESION Team/SEE tightly integrates leading CASE tools for systems analysis and design, programming, documentation, configuration management, project management, and other critical process functions into one complete, totally scalable and customizable solution. COHESION Team/SEE features the unique Process Assistant™ technology that promotes team-wide use of the user-defined software development process, and automates change control.

## COHESIONworX

**Product Information:**

Version Number: 2.0  
Date of Latest Release: 26 March 1994  
Date of First Release: October 1993  
Frequency of Updates:  
Number Sold:

**Pricing:**

Single User Price: \$1595 (per concurrent user)  
Site License:  
Multicopy Price: Digital volume discount  
agreements apply

GSA Price:  
Maintenance Price:

**Vendor Contact Information:**

Digital Equipment Corporation  
Address: 4 Results Way, MRO4-3/H6  
Marlboro, MA 01752

In Business Since: 1957

**Marketing Contact:** Andrew Willinger

Phone Number: 508-467-4041

Fax Number: 508-467-3106

E-mail Address: willinger@mr4dec.enet.dec

**Technical Contact:**

Phone Number:

Fax Number:

E-mail Address:

Bulletin Board System:

Customer Support:

**Platform(s)/Operating System(s):**

Alpha AXP / DEC OSF/1 AXP  
HP 9000 series / HP-UX  
SPARCstation / SUNOS

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

48/64 MB memory, 500 MB disk, Motif, DCE, TCPIP

**Description/Purpose:**

COHESIONworX combines a full-featured code development environment, utilities, graphical user interfaces, message passing technology, and multisystem integration. The result is an object-oriented, work-centered paradigm focused on productive software development. Additional tools may be easily integrated through encapsulation using the optional EnCASE toolkit. Based on SCO's popular IXLdesktop, COHESIONworX adds CORBA and DCE-based control services to transparently support a true multivendor network with the simplicity of drag-and-drop icons.

## Cradle Systems Engineering Environment (SEE)

**Product Information:**

Version Number: 1.8  
Date of Latest Release: Apr 1994  
Date of First Release: May 1989  
Frequency of Updates: 6 to 9 months  
Number Sold: ~500

**Pricing:**

Single User Price: Cradle consists of several functional units.  
Contact vendor.  
Site License: Contact vendor.  
Multicopy Price: Contact vendor.  
  
GSA Price: Contact vendor.  
Maintenance Price: 15 percent of list price.

**Vendor Contact Information:**

Mesa Systems Guild, Inc.  
Address: 168 Ninth Street  
Providence, RI 02906-2931

In Business Since: 1990

**Marketing Contact:** Michael R. Bousquet

Phone Number: 401-421-9390  
Fax Number: 401-421-3590  
E-mail Address: 73340.3066@compuserve.com

**Technical Contact:** Brian Brady

Phone Number: 401-421-9390  
Fax Number: 401-421-3590  
E-mail Address: 73340-3066@compuserve.com

Bulletin Board System: N/A

Customer Support: 401-421-9390

**Platform(s)/Operating System(s):**

DEC 2100, 3100, 5000, ULTRIX 4.2 or later.  
DEC VAXstation, OpenVMS 5.5 or later.  
HP/Apollo 9000/400, Domain SR10.3 or later.  
HP 9000/700, HP/UX 8.7 or later.  
IBM RS/6000, AIX 3.2 or later.  
PC's any 386/486, SCO UNIX Version 3.2v2, OpenDesktop 1.0, 1.1, and 2.0.  
Sun Sun/4 (all models), Sun OS 4.0.3 or later.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

RAM size 8 MB, disk storage space 24 MB, User Interface, X Windows, Network Ethernet TCP/IP, Swap Space 25 MB per Cradle application.

**Description/Purpose:**

Cradle/SEE is a complete life cycle support environment for the Systems and Software Engineer. Cradle SEE is fully integrated, multiuser, multiproject Systems Engineering Environment for turning formalized requirements into high quality systems and software. Cradle's engineering environment provides an automated life-cycle development process in an open and extensible framework. Cradle SEE Total System consists of Requirements Capture, Functional Analysis, Performance Analysis, Design, Design Evaluation, Code Generation, Reverse Engineering, Documentation, Configuration Management, Project Support/Metrics.

## EAST Environment

<b>Product Information:</b> Version Number: Contact vendor. Date of Latest Release: Date of First Release: Frequency of Updates: Number Sold: 600	<b>Vendor Contact Information:</b> Mark V Systems Address: 16400 Ventura Blvd., #303 Encino, CA 91436  In Business Since: 1974 <b>Marketing Contact:</b> Mo Bjornestad Phone Number: 818-995-7671 Fax Number: 818-995-4267 E-mail Address: mob@markv.com <b>Technical Contact:</b> Support Phone Number: 818-995-7671 Fax Number: 818-995-4267 E-mail Address:  Bulletin Board System: N/A Customer Support:
<b>Pricing:</b> Single User Price: Contact vendor Site License: Multicopy Price:   GSA Price: N/A Maintenance Price:	
<b>Platform(s)/Operating System(s):</b> SPARC/4.1 RS/6000 AIX 3.2	
<b>Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):</b> 32 - 48 MB RAM. 80 - 96 MB Swap. 400 MB Disk. X11 R4 - Motif 1.1.3.	
<b>Description/Purpose:</b> EAST - Environment provides essential services for the support of the development process, together with basic tool development support. Features include: Process Modeling, Process Management, Project Management, Documentation Generation, Configuration Management, Repository (PCTE) Management, User Assistance, Development tools for Ada, C, and FORTRAN and a Tool Interface Guide for encapsulation or integration of additional tools.	

## **Emeraude V12**

**Product Information:**

Version Number: 12.4  
Date of Latest Release: March 1993  
Date of First Release:  
Frequency of Updates: Annual  
Number Sold: 1,000 +

**Pricing:**

Single User Price: \$1,870 - \$6,860  
Site License: To be negotiated.  
Multicopy Price: Available. Contact vendor.

GSA Price:  
Maintenance Price: 10 percent.

**Vendor Contact Information:**

GIE Emeraude (Mark V Systems, U.S.A. distributor)

Address: Emeraude, BP3-PC6A1  
68 Route de Versailles  
78430 Louveclennes, France

In Business Since: 1983

**Marketing Contact:** Dominique Soufflet

Phone Number: +33 1 3902 4060

Fax Number: + 33 1 3902 4738

E-mail Address: Dominique.Soufflet@friv.bull.fr

**Technical Contact:** Regis Minot

Phone Number: +33 1 3902 4013

Fax Number: +33 1 3902 4206

E-mail Address: Regis.Minot@friv.bull.fr

Bulletin Board System:

Customer Support: Jonathan Jowett

**Platform(s)/Operating System(s):**

Sun SPARC, DEC 5000, Alpha, IBM RS/6000, HP 9000/700, Bull OPX 20.  
Sun OS 4.1, Solaris 2.1, ULTRIX + OSF/1, AIX 3.2, HP-UX, BOSX/3.2.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

Memory: 16 MB RAM.  
Disk: 100 MB Disk (servers only; diskless supported).  
User Interface: X/11, Motif, or OpenLook.  
Network: TCP/IP and NFS.

**Description/Purpose:**

Emeraude V12.4 is a framework on top of which Software Engineering Environments can be built and tools can be either integrated or encapsulated. It can also be used as a basic environment in itself.

## FreedomWorks

**Product Information:**

Version Number: 1.3  
Date of Latest Release: Feb 1994  
Date of First Release: Feb 1994  
Frequency of Updates: Semiannually  
Number Sold: 3 sites

**Pricing:**

Single User Price: \$3,300-\$10,000  
(Module price)  
Site License:  
Multicopy Price:  
  
GSA Price: Contact vendor.  
Maintenance Price: 20 percent of purchase price.

**Vendor Contact Information:**

Alslys, Inc.  
Address: 10251 Vista Sorrento Parkway  
San Diego, CA 92121

In Business Since: 1982

**Marketing Contact:** Bill Hart

Phone Number: 619-457-2700  
Fax Number: 619-452-1334  
E-mail Address: hart@alslys.com

**Technical Contact:**

Phone Number: 800-995-2579  
Fax Number:  
E-mail Address:

Bulletin Board System:

Customer Support:

**Platform(s)/Operating System(s):**

Sun Sparc	Sun OS
HP 9000/7XX	HP/UX
IBM RS/6000	AIX

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

32 MB RAM.  
250 MB Disk.  
Motif or OpenLook user interface.  
NFS if distributed development is desired.

**Description/Purpose:**

FreedomWorks is a modular family of products which form a scalable software engineering framework based on PCTE-compliant data repository. Starting with Object CM, a highly functional configuration management system, a user can incrementally build a complete software engineering environment with integrated project management, documentation management and software reuse capabilities. Support for tool integration and process definition is provided. A predefined 2167A model is included with the core product.

Add-on modules include Freedom/Documenter, which integrates documentation process into the software development environment. Freedom/Object Reuse is a library definition and browsing facility to assist in setting up a reuse database. Freedom/Project Manager is an integrated facility which allows managers to plan and allocate tasks, and obtain up-to-the-minute status information based on the current state of the software repository.

## Information Engineering Facility

**Product Information:**

Version Number: 5.1  
Date of Latest Release: Dec 1992  
Date of First Release: 1986  
Frequency of Updates: Semiannual  
Number Sold: 600 +

**Pricing:**

Single User Price: \$7,280 - \$17,290  
Site License:  
Multicopy Price: ~ \$125,000/6  
~ \$500,000/30

GSA Price: Available.  
Maintenance Price: 15 percent of purchase price  
annually after 1 yr. warranty  
period.

**Vendor Contact Information:**

Texas Instruments  
Address: 1400 S. Potomac Street, Suite 101  
Aurora, CO 80012

In Business Since: 1930

**Marketing Contact:** Betty Iura

Phone Number: 303-369-1329  
Fax Number: 303-755-8434  
E-mail Address:

**Technical Contact:** Barbie Stafford

Phone Number: 303-369-1327  
Fax Number: 303-775-8434  
E-mail Address:

**Bulletin Board System:**

Customer Support: 214-575-5900

**Platform(s)/Operating System(s):**

IBM compatible computers with MS-DOS, OS/2, and Windows.  
IBM mainframe computers with MVS; RS/6000, HP 9000, and SUN computers with UNIX.  
Workstations that operate under UNIX.  
Tandem and Sequent with UNIX.  
DEC with VMS.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

Dependent on platforms.

**Description/Purpose:**

Information Engineering Facility (IEF) is a repository-based integrated software engineering environment that enables the automatic generation of small to very large MIS applications. Because of the IEF's 100 percent code generation feature, targeting multiple language SQL and operating system processors, the resulting applications offer a high degree of portability.

As a result of its model-based paradigm, coupled with its completeness and consistency checking, system development and maintenance is greatly simplified. With the availability of the Business Design Facility in mid-1993, support for IDEF0 and IDEF1x, and Activity-based Costing, the entire DoD cycle will be supported within the integrated CASE product set.

## Integrated System Development

**Product Information:**

Version Number: 1.0  
Date of Latest Release: May 1993  
Date of First Release: May 1993  
Frequency of Updates: Semiannual  
Number Sold:

**Pricing:**

Single User Price: Pricing is based upon  
Site License: customer requirements.  
Multicopy Price: Contact vendor.

GSA Price:  
Maintenance Price:

**Vendor Contact Information:**

Bull HN Information Systems Inc.  
Address: 300 Concord Road  
Billerica, MA 01821

In Business Since: 1955

**Marketing Contact:** Steve Barry

Phone Number: 508-294-4572  
Fax Number: 508-294-2653  
E-mail Address: s.barry@bull.com

**Technical Contact:** Cameron McLean

Phone Number: 508-294-4623  
Fax Number: 508-294-3837  
E-mail Address: c.mclean@bull.com

Bulletin Board System: None  
Customer Support:

**Platform(s)/Operating System(s):**

Bull's DPX/20 UNIX - BOS/X Operating System.  
IBM's RS/6000 UNIX - AIX Operating System.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

RAM - 32MB.  
Disk - 400 MB.  
User Interface - X/11 R5, Motif.  
Network - TCP/IP.

**Description/Purpose:**

Integrated System Development (ISD) is comprised of a standards-based repository, a neutral information model, and a standards-based control integration technology together with professional services available from Bull for customization, tool integration, and implementation. ISD is built around a repository that is based on the ECMA-149 PCTE standard developed and licensed from GIE Emeraude; an organization jointly owned by Bull and Thomson/SYSECA and developers of the first commercially available PCTE repository.

## IPSYS ToolBuilder

**Product Information:**

Version Number: 1.3  
Date of Latest Release: Dec 1992  
Date of First Release: Third quarter, 1991  
Frequency of Updates: Semiannual  
Number Sold: N/A

**Pricing:**

Single User Price: Contact vendor.  
Site License:  
Multicopy Price:

GSA Price:  
Maintenance Price:

**Vendor Contact Information:**

IPSYS Software  
Address: 28 Green Street  
Newbury, MA 01951

In Business Since: 1989

**Marketing Contact:** Donna MacDonald

Phone Number: 508-463-0006

Fax Number: 508-462-9198

E-mail Address:

**Technical Contact:** Technical Support

Phone Number: 508-463-0006

Fax Number: 508-462-9198

E-mail Address:

Bulletin Board System:

Customer Support:

**Platform(s)/Operating System(s):**

SUN SPARC, SUN OS/SOLARIS.  
HP 9000, HP-UX.  
IBM RS/6000 AIX.  
DECstation, ULTRIX.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

RAM: 16 MB, swap space: 40 MB.  
Disk space: 25 MB, User Interface: Motif, OpenLook, X Window.  
Installation Media: QIC/DAT.  
Network: Ethernet/UNIX Networking, Postscript (or compatible) printer.

**Description/Purpose:**

IPSYS ToolBuilder is a meta-CASE tool, i.e., a CASE tool used to build CASE and application development tools. These tools may support a standard or an organization-specific method. Tools can be built quickly and enhanced or customized quickly. CASE tools developed using IPSYS ToolBuilder can support the software development life cycle from planning and analysis to design, code generation, and automatic and complete design document generation. Other important facilities include a PCTE-compliant repository and full on-line multiuser working.

## LIFE\*CYCLE

**Product Information:**

Version Number: 2.2  
Date of Latest Release: Nov 1993  
Date of First Release: Mar 1992  
Frequency of Updates: Annually  
Number Sold: 300 +

**Pricing:**

Single User Price: \$7,500  
Site License: Available.  
Multicopy Price: Available.

GSA Price: Available.  
Maintenance Price: 18 percent.

**Vendor Contact Information:**

CRI Inc.  
Address: 3245 146th Place, SE #270  
Bellevue, WA 98007

In Business Since: 1979

**Marketing Contact:** John Wilkerson

Phone Number: 206-643-7443

Fax Number: 206-643-6873

E-mail Address:

**Technical Contact:** Andy Bliven

Phone Number: 206-643-7443

Fax Number: 206-643-6873

E-mail Address: bliven@criinc.com

Bulletin Board System: 206-643-7443

Customer Support:

**Platform(s)/Operating System(s):**

Sun Microsystem.  
IBM RS-6000/AIX.  
HP-UX.  
Sun OS and Solaris 2.3.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

RAM - 32 MB/48 MB.  
Disk - Serves 500 MB/1GB.  
Network - TCP/IP.  
User Interface - Motif, OpenLook.

**Description/Purpose:**

LIFE\*CYCLE is a fully integrated Software Development Environment support the whole software life cycle.  
LIFE\*CYCLE encompasses the use of leading edge technologies and methodologies.

### Metatool Specification Driver

**Product Information:**

Version Number: 2.0  
Date of Latest Release:  
Date of First Release:  
Frequency of Updates:  
Number Sold:

**Pricing:**

Single User Price: \$15,000  
Site License: N/A  
Multicopy Price: \$65,000

GSA Price: N/A  
Maintenance Price: 12 percent of initial  
license fee.

**Vendor Contact Information:**

AT&T  
Address: 10 Independence Blvd.  
Warren, NJ 07059

In Business Since: 1890  
**Marketing Contact:** Ed Cartier  
Phone Number: 908-580-5719  
Fax Number: 908-580-6355  
E-mail Address: attmail!ecartier  
**Technical Contact:** Kristin Kassas  
Phone Number: 508-960-6473  
Fax Number:  
E-mail Address:

Bulletin Board System:  
Customer Support:

**Platform(s)/Operating System(s):**

Unix Operating System Release 5, Version 3 or higher, including a C compiler, MAKE, LEX and YACC.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

1 MB disk space, 1 MB RAM, 2 MB per user file.

**Description/Purpose:**

Metatool is a software tool for building other software tools such as translators, compilers, and application generators. The output (software tools) of Metatool is generated from high-level specifications that are translated by the software into machine-readable code.

## METHODMANAGER

**Product Information:**

Version Number: 2.7.0  
Date of Latest Release: 1992  
Date of First Release: 1975  
Frequency of Updates:  
Number Sold:

**Pricing:**

Single User Price:  
Site License:  
Multicopy Price:

GSA Price:  
Maintenance Price:

**Vendor Contact Information:**

Manager Software Products, Inc.  
Address: 131 Hartwell Avenue  
Lexington, MA 02173

In Business Since: 1966

**Marketing Contact:** Carl Olofson

Phone Number: 617-863-5800  
Fax Number: 617-861-6130  
E-mail Address: 71431.2032@compuserve.com

**Technical Contact:** Carl Olofson

Phone Number:  
Fax Number:  
E-mail Address:

Bulletin Board System:  
Customer Support:

**Platform(s)/Operating System(s):**

IBM MCA/TSO.  
IBM VM/CMS.  
Microsoft Windows 3.1 +.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

HOST: 370-type mainframe, MVS or VM/CMS, approximately 100 cylinders.  
Workstation: 4-6 MB RAM MS-Windows 3.1+, 4-12 MB Disk, Mouse.

**Description/Purpose:**

## Objectime

**Product Information:**

Version Number: 4.1.1.  
Date of Latest Release: Feb 1994  
Date of First Release: Feb 1991  
Frequency of Updates: 6 months  
Number Sold: 300

**Pricing:**

Single User Price: \$20,000  
Site License: To be negotiated.  
Multicopy Price: Sliding scale \$20,000 to  
\$10,000 depending on  
number of copies.  
  
GSA Price: N/A  
Maintenance Price: \$3,000 a year with first year  
included in purchase price.

**Vendor Contact Information:**

Objectime Limited  
Address: 340 March Road, Suite 200  
Kanata, Ontario Canada K2K 2E4

In Business Since: 1991

**Marketing Contact:** Garth Gullekson

Phone Number: 800-567-TIME  
Fax Number: 613-591-3784  
E-mail Address: sales @objectime.on.ca

**Technical Contact:** Technical Support

Phone Number: 613-591-3400  
Fax Number: 613-591-3784  
E-mail Address: support@objedtime.on.ca

Bulletin Board System:  
Customer Support:

**Platform(s)/Operating System(s):**

Sun SPARC, HP 425 and IBM RS/6000. Contact vendor for status on other platforms.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**  
16 MB RAM/24-28 MB Disk.

**Description/Purpose:**

Objectime is an object-oriented CASE tool for real-time systems supporting the Real-time Object-oriented Modeling (ROOM) methodology and enabling the creation of executable analysis and design models.

## ObjectMaker

**Product Information:**

Version Number: 3.0  
Date of Latest Release: April 1994  
Date of First Release: 1986  
Frequency of Updates: 6 months  
Number Sold: 1,500 +

**Pricing:**

Single User Price: \$1,250 to \$5,000  
Site License: Contact vendor.  
Multicopy Price: Contact vendor.

GSA Price: N/A  
Maintenance Price: Contact vendor.

**Vendor Contact Information:**

Mark V Systems  
Address: 16400 Ventura Blvd., #303  
Encino, CA 91436

In Business Since: 1974

**Marketing Contact:** Mo Bjornestad

Phone Number: 818-995-7671

Fax Number: 818-995-4267

E-mail Address: mob@markv.com

**Technical Contact:** Support

Phone Number:

Fax Number:

E-mail Address:

Bulletin Board System:

Customer Support: Q21994

**Platform(s)/Operating System(s):**

HP, IBM, Sun, Mips, SGI, Motorola, Data General, SCO, DEC, NCR,  
DOS, NT, Unix, VMS, OS/2, MS Windows, X Windows, DECwindows.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**  
4.8 MB RAM, 10 MB Disk.

**Description/Purpose:**

Graphical modeling tool supporting 30 traditional and object-oriented notations/methods for analysis and design. Underlying repository captures semantics in a unified abstract Object Model. Transformations to and from code (Ada, C and C++). Metatool version available.

## Objectory

<b>Product Information:</b> Version Number: 3.3.1 Date of Latest Release: Feb 1993 Date of First Release: June 1988 Frequency of Updates: 2 years Number Sold: 450	<b>Vendor Contact Information:</b> Objective Systems Address: (U.S. Subsidiary) 4 Greenwich Office Park Greenwich, CT 06831  In Business Since: 1987 <b>Marketing Contact:</b> Per Kroll Phone Number: 203-625-7250 Fax Number: 203-625-7272 E-mail Address: <b>Technical Contact:</b> Per Kroll Phone Number: 203-625-7250 Fax Number: 203-625-7272 Bulletin Board System: mserv@os.se Customer Support: Johan Naesman Phone Number: +46 8 703 4572 Fax Number: +46 8 751 3096
<b>Pricing:</b> Single User Price: \$4,000 - \$10,000 Site License: On request. Multicopy Price:   GSA Price: Maintenance Price: 1 year included.	

### Platform(s)/Operating System(s):

All major Unix platforms.  
IBM PC (Windows).  
DOS/Windows.  
OS/2.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**  
16 MB RAM; 25 MB Disk.

### Description/Purpose:

Objectory Support Equipment (SE) is an object-oriented software system implemented in Smalltalk. It is a multiuser system that can be installed in a network to support system development teams. In a complete support environment, Objectory SE is combined with complementary tools, such as programming environments, configuration management and project management tools.

## OOSD/Ada - Software Through Pictures

**Product Information:**

Version Number: 1.2  
Date of Latest Release: 1992  
Date of First Release: 1990  
Frequency of Updates: Annually  
Number Sold: 700 +

**Pricing:**

Single User Price: \$8,500  
Site License: Contact vendor.  
Multicopy Price: Contact vendor.

GSA Price: Contact vendor.  
Maintenance Price:

**Vendor Contact Information:**

IDE  
Address: 595 Market Street, 10th Floor  
San Francisco, CA 94105

In Business Since: 1983

**Marketing Contact:** Chris Lalli  
Phone Number: 415-543-0900  
Fax Number: 415-543-0145  
E-mail Address: Chris@ide.com

**Technical Contact:**

Phone Number: 800-444-7871  
Fax Number: 415-543-1314  
E-mail Address: techstaff@ide.com

Bulletin Board System: Yes

Customer Support: Yes

**Platform(s)/Operating System(s):**

Sun - SPARCstation.  
HP 9000 - 700/800 - HP-UX.  
DECstation - ULTRIX.  
IBM - RS/6000 AIX.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

16 MB RAM; 35 MB Disk; X11, DECwindows, OpenWindows; Ethernet.

**Description/Purpose:**

Object-Oriented Software Design (OOSD)/Ada consists of a family of graphical editors integrated through a data dictionary that enables a software engineer to diagrammatically depict architectural representations and detailed design representations of an Ada system. The editors support widely recognized notations from Constantine/Yourdon, Booch and Hoard. The tool supports automatic generation of 2167A documentation and has an optional code generator and Reverse Engineering tool.

## PCTE Workbench

**Product Information:**

Version Number: 1.1  
Date of Latest Release: Oct 1993  
Date of First Release: Oct 1992  
Frequency of Updates: 1 to 2 per year.  
Number Sold:

**Pricing:**

Single User Price: \$5,000  
Site License: Available.  
Multicopy Price: Discounts available on multiple copies.

GSA Price:  
Maintenance Price: \$750/year.

**Vendor Contact Information:**

Vista Technologies, Inc.  
Address: 1100 Woodfield Rd, Suite 437  
Schaumburg, IL 60173-5124

In Business Since: 1987

**Marketing Contact:** Edith Ludwig  
Phone Number: 708-706-9300  
Fax Number: 708-706-9317  
E-mail Address: info@vistatech.com

**Technical Contact:** David Hurst  
Phone Number: 708-706-9300  
Fax Number: 708-706-9317  
E-mail Address: hurst@vistatech.com

Bulletin Board System:  
Customer Support: info@vistatech.com

**Platform(s)/Operating System(s):**

Sun SPARCstation/Sun OS 4.1X  
IBM RS/6000 IBM AIX 3.2

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**  
16 MB RAM, 50 MB disk, X Window system.

**Description/Purpose:**

PCTE Workbench is a toolkit for building hypermedia based environments and applications using PCTE. It provides easy-to-use tools for browsing, building, editing and querying the PCTE repository and a sophisticated tool composition mechanism enabling applications to be built quickly and easily.

## Process Weaver

**Product Information:**

Version Number: 2.0  
Date of Latest Release: Feb 1994  
Date of First Release: June 1992  
Frequency of Updates: 6 months  
Number Sold: 70

**Pricing:**

Single User Price: Client \$835 +  
Site License: Server: \$16,800  
Multicopy Price: Quantity discounts available.

GSA Price: N/A  
Maintenance Price: 13 percent

**Vendor Contact Information:**

Cap Gemini Innovations  
Address: Avenue of the Americas  
29th Floor  
New York, NY 10036

In Business Since: 1980

**Marketing Contact:** Larry Proctor

Phone Number: 915-947-7183

Fax Number: 915-942-9496

E-mail Address:

**Technical Contact:** Vincent Jorrano

Phone Number: 1-800-400-1511

Fax Number:

E-mail Address:

Bulletin Board System: 1-800-400-1511

Customer Support: 8 a.m. to 6 p.m. EST

**Platform(s)/Operating System(s):**

Unix on IBM RS/6000, Sun-4, HP 700, DEC 5000.  
Support for mixed environments available for clients and servers.  
MS Windows 3.1 supported by client.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

Unix: 16 MB memory, 50 MB disk, OSF/Motif, X Windows, TCP/IP.

**Description/Purpose:**

Process modeling and enactment allows users to: model methods, procedures and organizations; provide customization and end-user support; interface third party CASE tools or applications; distributed on Local Area Network (LAN) or Wide Area Network (WAN).

### Pro-IV Workbench

**Product Information:**

Version Number: 1.0  
Date of Latest Release: Oct 1993  
Date of First Release: Oct 1988  
Frequency of Updates: Semiannual  
Number Sold: 4,500 +

**Pricing:**

Single User Price: \$1,495  
Site License: N/A  
Multicopy Price: 6-10 cys, 5 percent  
11-15 cys, 10 percent  
16-19 cys, 15 percent  
20-25 cys, 20 percent  
26 + cys, 25 percent  
GSA Price: 15 percent List Price.  
Maintenance Price: Contact vendor.

**Vendor Contact Information:**

McDonnell Douglas Information Systems Ltd.  
Address: 325 J.S. McDonnell Blvd.  
Mailcode 3065600  
Hazelwood, MO 63042

In Business Since: CASE since 1982

**Marketing Contact:** Al Weiss

Phone Number: 314-233-6068  
Fax Number: 314-233-6331  
E-mail Address:

**Technical Contact:**

Phone Number: 800-225-7760  
Fax Number: 314-233-6331  
E-mail Address:

Bulletin Board System: Yes

Customer Support: 800-225-7760

**Platform(s)/Operating System(s):**

DOS 3.3 or higher.  
Windows 3.1.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

Intel 80386 or 80486, 2 MB or RAM, 80 MB hard drive, Windows 3.1 or DOS.  
Multiuser version requires LAN.

**Description/Purpose:**

PRO-IV Workbench for MS Windows is a set of advanced, integrated development tools for the analysis, design, and development phases of the software life cycle. A repository manages the data and graphics in an object-oriented environment. The product is available as single user or a real-time client/server version that allows two or more developers to share the same project information.

## ProSLCSE

**Product Information:**

Version Number: Alpha  
Date of Latest Release: Feb 1994  
Date of First Release: Apr 1994  
Frequency of Updates: 6 months  
Number Sold:

**Pricing:**

Single User Price: GOTS  
Site License:  
Multicopy Price:

GSA Price:  
Maintenance Price: Contact vendor.

**Vendor Contact Information:**

International Software Systems, Inc.  
Address: 5430 Research Blvd.  
Echelon IV, Suite 250  
Austin, TX 78759

In Business Since: 1979

**Marketing Contact:** Priscilla Yeh

Phone Number: 512-338-5721

Fax Number: 512-338-5757

E-mail Address:

**Technical Contact:**

Phone Number:

Fax Number:

E-mail Address:

Bulletin Board System:

Customer Support:

**Platform(s)/Operating System(s):**

SPARC 1/SPARC 2.

Sun/Sun OS 4.X.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

28 MB RAM.

Two 105 MB Hard Drives.

OpenLook, UI/Windows Manager.

Ethernet.

**Description/Purpose:**

The Process Oriented Software Life Cycle Support Environment (ProSLCSE) will be designed to assist organizations to: define, enact, and improve their Software Development Process.

**SoftBench, SoftBench/C++**

**Product Information:**

Version Number: 3.2  
Date of Latest Release: Jan 1994  
Date of First Release: Jan 1990  
Frequency of Updates: Biannually  
Number Sold: >25,000

**Pricing:**

Single User Price: SoftBench \$2,300  
C++ SoftBench \$4,500  
Cobol SoftBench \$2,795  
Site License: N/A  
Multicopy Price: Determined by contract.  
GSA Price: Avail for project budgeting  
and/or purchase.  
Maintenance Price:

**Vendor Contact Information:**

Hewlett-Packard Company (SESD)  
Address: 3404 East Harmony Road  
Ft. Collins, CO 80524

In Business Since: 1939

**Marketing Contact:** Chuck Dahl

Phone Number: 206-643-8784  
Fax Number: 206-643-8743  
E-mail Address: Chuck\_dahl@hpatc3.desk.hp.com

**Technical Contact:** Michale Barrat

Phone Number: 206-644-3253  
Fax Number: 206-643-8748  
E-mail Address: mikeb@hpubvwa.nsr.hp.com

Bulletin Board System:  
Customer Support:

**Platform(s)/Operating System(s):**

HP 9000/HP-UX.  
Sun-SPARC/Sun OS.  
SoftBench is licensed to other system suppliers and runs on IBM/AIX; CDC/MIP; Siemens Nixdorf, and Fujitsu.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

RAM = 24 MB.  
Disk Space = SoftBench = 9 MB; C++ SoftBench = 12 MB (with compiler and LIB's = 48 MB).  
User Interface: X/11 Motif or Open Look on Sun.

**Description/Purpose:**

Hewlett-Packard's (HP) SoftBench has two major components: First, as a framework it is an enabling technology. The framework supports inter-tool communication, distributed computing and a consistent user interface. Software tools can be integrated without source code changes using HP's encapsulator. Second, SoftBench is an integrated construction tool set (Edit, analyze, debug, compile) supporting software development for C++, C, FORTRAN, COBOL, and Pascal.

## Software Backplane

**Product Information:**

Version Number: 3.0.2  
Date of Latest Release: Sept 1993  
Date of First Release: Mar 1988  
Frequency of Updates: 6 months  
Number Sold: 1300 seats

**Pricing:**

Single User Price: \$3,995  
Site License: To be negotiated.  
Multicopy Price: Discounts based on dollar volume.  
  
GSA Price: N/A  
Maintenance Price: 18 percent of product price/year.

**Vendor Contact Information:**

Atherton Technology  
Address: 39350 Civic Center Drive  
Suite 430  
Fremont, CA 94538

In Business Since: 1986

**Marketing Contact:** Jim Conlon

Phone Number: 510-494-8411

Fax Number: 510-494-1776

E-mail Address: jim@atherton.com

**Technical Contact:** Chan Drawdy

Phone Number: 510-494-8411

Fax Number: 510-494-1776

E-mail Address: chan@atherton.com

Bulletin Board System: USENET

Customer Support: 510-494-8411

support@atherton.com

**Platform(s)/Operating System(s):**

Sun 4 Family on Sun OS.  
DECstation ULTRIX.  
IBM AIX workstations on Unix.  
HP Series 700.  
SGI IRIX.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

RAM: 16 MB, Network: Ethernet, TCP/IP NFS.  
Disk: 150 MB.  
GUI: X Window, Motif.  
OTHER: SCCS; Swapfile - Unix 40 MB.

**Description/Purpose:**

Software Backplane is an Open Systems Software Development Repository based on a distributed, heterogeneous object database; objects stored consist of data and behavior. Included in the tool is a library of object types which provide comprehensive configuration management, tool integration, and process integration systems to support the building of a software development environment. In 1993 a Reuse Management System was added which enabled developers to identify and reuse high quality, error-free components.

## STRAND-RT - Software Through Pictures

**Product Information:**

Version Number: 4.2D  
Date of Latest Release: 1992  
Date of First Release: 1988  
Frequency of Updates: Yearly  
Number Sold: 10,000 +

**Pricing:**

Single User Price: \$15,000  
Site License: Contact vendor.  
Multicopy Price: Contact vendor.

GSA Price: \$6,750.06/user  
Maintenance Price: \$1,014/year

**Vendor Contact Information:**

IDE

Address: 595 Market Street, 10th Floor  
San Francisco, CA 94105

In Business Since: 1983

**Marketing Contact:** Chris Lalli

Phone Number: 415-543-0900

Fax Number: 415-543-0145

E-mail Address: mray@ide.com

**Technical Contact:** Tech Support

Phone Number: 800-444-7871

Fax Number: 415-543-1314

E-mail Address: techstaff@ide.com

Bulletin Board System: Yes

Customer Support: Yes

**Platform(s)/Operating System(s):**

Sun-SPARCstation - Sun/OS and Solaris; HP 9000 - 300/400/700/800 - HP-UX;  
IBM RS/6000-AIX; DEC-DECstation - ULTRIX; DEC VAXstation - VMS; DEC Alpha-OSF1  
Intel 486 - SCO-Unix; NCR-Unix; DG-DG/UX; Motorola 88K - Sys V, Rel. 4;  
SGI - Irix; ApolloDNxxx, Domain OS; Ceetia - Unigraph; Sun 3 - Sun OS.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

16 MB for most machines; 12 MB for some machines;  
Disk Requirements - 28 MB min., 50 MB worst case;  
User Interface support - X11, OpenWindows, OpenLook; Motif Ethernet.

**Description/Purpose:**

A family of graphical editors links a data directory that supports a variety of software and Systems Engineering Methodologies. These graphical editors support Yourdon Demarco or Gane Sarson Structured Analysis; Jackson Method for defining data structures; Constantine/Yourdon Structured Design; and Hatley/Pirbhai Real Time Systems Specification. Code Generation and 2167 and 2167A documentation are outputs of the data dictionary.

**VSF, MWB-SF/AWB-SF**

**Product Information:**

Version Number: 3.8  
Date of Latest Release: Jan 1993  
Date of First Release: Nov 1987  
Frequency of Updates: Semiannual  
Number Sold:

**Pricing:**

Single User Price: \$6,000/\$25,000  
Site License: \$177,000  
Multicopy Price:

GSA Price: N/A  
Maintenance Price: 17.5 percent

**Vendor Contact Information:**

Virtual Software Factory Limited  
Address: 13873 Park Center Road, Suite 218  
Herndon, VA 22071

In Business Since: 1986

**Marketing Contact:** Steve Shap

Phone Number: 703-318-1180

Fax Number: 703-318-1190

E-mail Address:

**Technical Contact:** Charles Williams

Phone Number: 703-318-1180

Fax Number: 703-318-1190

E-mail Address:

Bulletin Board System:

Customer Support:

**Platform(s)/Operating System(s):**

RS/6000/AIX.  
PS/2 and 486/OS/2.  
VAXstation/VMS.  
DECstation/ULTRIX.  
SUN SPARC/UNIX.

**Minimum/Recommended Configuration (RAM size, disk storage space, user interface, network, etc.):**

12 MB RAM, 40 MB swap space, 120 MB disk.  
Motif on Unix.  
DECWindows on VMS.  
Presentation Manager on OS/2.

**Description/Purpose:**

The Virtual Software Factory is truly unique within the CASE world market being a workbench for CASE tool builders and a range of specific CASE products that can be used either as self-contained tools or as fragments for developing bespoke solutions.

The Method WorkBench-Software Factory is a configurable CASE tool environment and technology. The Analysis WorkBench -Software Factory is the run-time CASE tool.

## **Appendix E: SEE Product Critiques**

## SEE Product Critique

**Product Name:** AISLE - Ada Environment  
**Vendor Name:** Software Systems Design  
**Special Operating Environment:**

**Version Number:** 5.4.1  
**Hardware Platform:** VAX 3100-10E  
**Operating System:** VMS 5.4-3

**Project Description:**  
USAF satellite program

**Years of software experience:** 15  
**Years of experience with product:** 3  
**Last time product was used:** Currently

**Overall Impression of this Product:**  
Good  
**Quality of Vendor Support:**  
Excellent

**Notable Strengths of this Product:**

- Ability to handle "huge" amounts of source code.
- If the software maintenance is purchased, you will get the best software maintenance cooperation I have ever encountered.

Note: We only use the "Ada DL" portion of the AISLE toolset at this time and only to support Reverse Engineering of contractor-supplied Ada source code for IV&V.

**Notable Weaknesses of this Product:**

Problems with some "overloaded" object names—will be addressed in a future major release.

**Advice for Potential Users or Buyers of this Product:**

If used on VAX/VMS, you will need to enhance the supplied "package standard" with the VAX Ada extensions (ACS extract source).

**Vendor Comments:**

## SEE Product Critique

**Product Name:** AISLE - Ada Environment

**Vendor Name:** Software Systems Design

**Special Operating Environment:**

**Version Number:** 5.3.N/3.6.C

**Hardware Platform:** VAX Silicon

Graphics/Convex

**Operating System:** VMS/Unix

**Project Description:**

Extended Air Defense Testbed

**Years of software experience:** 14

**Years of experience with product:** 4

**Last time product was used:** July 1992

**Overall Impression of this Product:**

Good

**Quality of Vendor Support:**

Excellent

**Notable Strengths of this Product:**

Good Ada automated documentation generation.

Note: Ada DL and Docgen are the portions of the AISLE toolset used by this group.

**Notable Weaknesses of this Product:**

Poor mixed language (Ada & C) capability.

**Advice for Potential Users or Buyers of this Product:**

Not a design tool. Ada DL and Docgen are only part of the vendor's life cycle tool complement. Plan how Ada DL and Docgen are to be used before starting.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** IEF  
**Vendor Name:** Texas Instruments  
**Special Operating Environment:**

**Version Number:** 4.x  
**Hardware Platform:** IBM  
**Operating System:** AVS

**Project Description:**

System development for loan tracking, accounts payable, mailing list, etc.

**Years of software experience:** 16  
**Years of experience with product:** 5  
**Last time product was used:**

**Overall Impression of this Product:**  
Excellent  
**Quality of Vendor Support:**  
Excellent

**Notable Strengths of this Product:**

- Excellent integration.
- Quality code is generated.
- Improves productivity.
- It does what the vendor says it will do.

**Notable Weaknesses of this Product:**

None.

**Advice for Potential Users or Buyers of this Product:** (from analysis through generation )

Get a hands-on demonstration of the product then try to do the same thing with any other products you are evaluating.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** Life\*Cycle

**Vendor Name:** Computer Resources International

**Special Operating Environment:** Atherton Backplane, Sun

**Version Number:**

**Hardware Platform:** Sun, IBM  
SPARC, RS/6000

**Operating System:** Sun OS, AIX

**Project Description:**

Computer Aided Project Engineering: Develop/Acquire a system/software engineering environment.

**Years of software experience:** 20

**Years of experience with product:** 1

**Last time product was used:**

**Overall Impression of this Product:**

Good

**Quality of Vendor Support:**

Excellent

**Notable Strengths of this Product:**

- Integrates activities of system engineers and software engineers in a common, repository-based environment.
- Implements and supports a defined process.
- Manages project data, i.e., version control, configuration management, access control.
- Traceability across whole life cycle.
- Tailor able to company/project standards.

**Notable Weaknesses of this Product:**

- Requires large amounts of RAM and virtual memory.
- Performance can be slow at times.
- Underlying database is not COTS.
- Requires expert support to administer the environment.
- Does not yet support distributed database.

**Advice for Potential Users or Buyers of this Product:**

Understand and document your development process and be prepared to either tailor the Life\*Cycle environment or accept the implied process of the environment.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** ObjectMaker  
**Vendor Name:** Mark V Systems Limited  
**Special Operating Environment:**

**Version Number:** 2.0.3  
**Hardware Platform:** VAXstation 3100  
**Operating System:** VMS 5.5-2

**Project Description:**

Graphic design capture from static analysis repository and code generation from design.

**Years of software experience:** 5  
**Years of experience with product:** 2  
**Last time product was used:** Currently

**Overall Impression of this Product:**  
Good  
**Quality of Vendor Support:**  
Good

**Notable Strengths of this Product:**

- It is very easy to extend the product.
- Built-in rules exist to create new methodologies.
- An extension language is available to customize code and report generating.
- Runs on multiple platforms and the diagrams and repository are compatible across platforms.

**Notable Weaknesses of this Product:**

Documentation is minimal. There is little on-line help. Diagrams are linked to repository through mapping process, which can take a long time during initial mapping. Found some problems with extension language when waiting code generation rules.

**Advice for Potential Users or Buyers of this Product:**

Forward engineering software worked well on all platforms; however, reverse engineering medium-to-large systems required workstations with as much memory and the fastest processor available.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** Objectime  
**Vendor Name:** Objectime Ltd.  
**Special Operating Environment:**

**Version Number:** 4.0  
**Hardware Platform:** Sun  
**Operating System:** Sun OS 4.1.3

**Project Description:**  
Telephone network prototyping

**Years of software experience:** 19  
**Years of experience with product:** 1.5  
**Last time product was used:** Dec 1992

**Overall Impression of this Product:**  
Excellent  
**Quality of Vendor Support:**  
Excellent

**Notable Strengths of this Product:**  
- Able to develop quick prototypes for analysis, connectivity to C++.  
- Run-time performance.

**Notable Weaknesses of this Product:**  
No comment.

**Advice for Potential Users or Buyers of this Product:**  
No comment.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** OOSD/Ada Software Through Pictures  
**Vendor Name:** IDE  
**Special Operating Environment:**

**Version Number:** 1.2  
**Hardware Platform:** SPARC10  
**Operating System:** Sun OS 4.1.3

**Project Description:**  
Satellite Decomutation hardware and software upgrade.

**Years of software experience:** 3  
**Years of experience with product:** < 1  
**Last time product was used:** Current

**Overall Impression of this Product:**  
Good  
**Quality of Vendor Support:**  
Fair

**Notable Strengths of this Product:**  
Liked CodeGen and Docgen DPS. Integration is good, but flexibility is poor.

**Notable Weaknesses of this Product:**  
- DPS template is awkward.  
- CodeGen format not flexible.  
- Not Motif-compliant.  
- Too many windows popping up all the time.

**Advice for Potential Users or Buyers of this Product:**  
Wait for next release.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** OOSD/Ada Software Through Pictures  
**Vendor Name:** IDE  
**Special Operating Environment:**

**Version Number:** 1.2  
**Hardware Platform:** SPARC2  
**Operating System:** Sun OS

**Project Description:**

Replace current Army Text and Experimentation System with mobile system.

**Years of software experience:** 28  
**Years of experience with product:** 1  
**Last time product was used:** Current

**Overall Impression of this Product:**  
Good  
**Quality of Vendor Support:**  
Excellent

**Notable Strengths of this Product:**  
Like the structured analysis features.

**Notable Weaknesses of this Product:**

- Immature code generation.
- No link from structured analysis to object-oriented design.
- Hard copy of process narrative (annotation) not very meaningful.

**Advice for Potential Users or Buyers of this Product:**

Cut through the hyperbole and find out exactly what the product will do and not do in a practical development environment. Good product and getting better, but best to know problems before rather than find out the hard way.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** ProKit\*Workbench  
**Vendor Name:** McDonnell Douglas Inf. Systems  
**Special Operating Environment:**

**Version Number:** 2.2  
**Hardware Platform:** 386 PCs  
**Operating System:** MS-DOS 5.0

**Project Description:**

Payload Data Management System—a series of subsystems that are used to process payload information, i.e., procurement, scheduling, engineering drawings.

**Years of software experience:** 9  
**Years of experience with product:** 6  
**Last time product was used:** Current

**Overall Impression of this Product:**  
Good  
**Quality of Vendor Support:**  
Fair

**Notable Strengths of this Product:**

Good analysis tool and data modeling. Great graphical output.

**Notable Weaknesses of this Product:**

Limited third party interfaces for code generation. Ad hoc reporting needs work, but is a good start for analyst documentation.

**Advice for Potential Users or Buyers of this Product:**

Ensure that a methodology is in place and that what this tool offers fits into your methodology. It can work if you want it to.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** ProKit\*Workbench  
**Vendor Name:** McDonnell Douglas Inf. Systems  
**Special Operating Environment:**

**Version Number:**  
**Hardware Platform:** 486 DX PC  
**Operating System:** DOS/Windows

**Project Description:**  
CASE Development System

**Years of software experience:** 10  
**Years of experience with product:** 7  
**Last time product was used:** Current

**Overall Impression of this Product:**  
Excellent  
**Quality of Vendor Support:**  
Excellent

**Notable Strengths of this Product:**  
Very complete, fully featured, well-integrated, easy to use, and well-supported.

**Notable Weaknesses of this Product:**  
None.

**Advice for Potential Users or Buyers of this Product:**  
Easiest to use, and most intuitive of standard methodology CASE tools.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** Rational APSE

**Vendor Name:** Rational

**Special Operating Environment:** Proprietary H/W and OS

**Version Number:** Delta 3.1

**Hardware Platform:** R1000 series

**Operating System:** Proprietary

**Project Description:** Mobile command and control system development.

**Years of software experience:** 25

**Years of experience with product:** 3

**Last time product was used:** Current

**Overall Impression of this Product:**

Excellent

**Quality of Vendor Support:**

Excellent

**Notable Strengths of this Product:**

- Ada.
- Integrated development environment with reasonable interfaces for integration with other CASE tools.
- Fine-grained information repository.
- Universal host supporting multiple target Ada compilers.
- Excellent CM/VC capabilities.

**Notable Weaknesses of this Product:**

Proprietary hardware.

**Advice for Potential Users or Buyers of this Product:**

Ensure adequate training is obtained as the environment has many features. Obtain implementation assistance from Rational.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** Software Backplane  
**Vendor Name:** Atherton Technology  
**Special Operating Environment:**

**Version Number:** 3.0  
**Hardware Platform:** Sun  
**Operating System:**

**Project Description:**

Build Program Support Environment for software engineers.

**Years of software experience:** 14  
**Years of experience with product:** 2  
**Last time product was used:** Current

**Overall Impression of this Product:**  
Fair  
**Quality of Vendor Support:**  
No comment

**Notable Strengths of this Product:**

Fast and powerful repository and methods. Lots of customization capabilities.

**Notable Weaknesses of this Product:**

The user interface is extremely difficult to learn and use. Integration of COTS products needs to be made easier.

**Advice for Potential Users or Buyers of this Product:**

You must have a team of engineers to support the customization of product. It is powerful, but there is a lot of "U" ware.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** Software Backplane

**Version Number:** 3.0

**Vendor Name:** Atherton Backplane

**Hardware Platform:** Sun, DEC, others

**Special Operating Environment:** 100 MB disk, 3 MB swap

**Operating System:** SunOS

**Project Description:** CORCASE: Development of integrated CASE toolset environment with value-added proprietary Process Manager.

**Years of software experience:** 25

**Overall Impression of this Product:**

**Years of experience with product:** 2 1/2

Excellent

**Last time product was used:** Current

**Quality of Vendor Support:**

Excellent

**Notable Strengths of this Product:**

Existing type hierarchy and inheritance capability, object-oriented repository with access control. Ability to refine existing methods and to create new types, methods/messages, and instance variables. Ease of version control and configuration management.

**Notable Weaknesses of this Product:**

Incomplete documentation in some areas. Incomplete functionality, but only in "obscure" cases.

**Advice for Potential Users or Buyers of this Product:**

Know your requirements for a SEE before making any selection, then determine if SBP satisfies your requirements.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** STRAND-RT Software Through Pictures

**Version Number:** 4.2D

**Vendor Name:** IDE

**Special Operating Environment:**

**Hardware Platform:** HP 720, HP 425

**Operating System:** HP-UX 8.\*

**Project Description:** High Altitude Missile Interception/Detection

**Years of software experience:** 12

**Years of experience with product:** 1

**Last time product was used:** Aug 1992

**Overall Impression of this Product:**

Excellent

**Quality of Vendor Support:**

Excellent

**Notable Strengths of this Product:**

Very open; existing networking infrastructure used; uses Hyland license server. Product allows extensive customization (which we utilized) and allows users to use existing editors with which they are familiar. Hatley Pirbhai method is well-supported except for architecture model (not supported). We regenerated our database every night in 90 minutes. Rebalancing and checking took two hours.

**Notable Weaknesses of this Product:**

The product was not integrated with IDE's Design Tool, SSD. Transition requires some manual steps. Like any tool operating on a large database (>5500 objects in our case), it requires workstation speed.

**Advice for Potential Users or Buyers of this Product:**

Use a fast processor (>30 SPEC) for rebalancing the entire database. Stick to Unix hosts such as Sun or HP; avoid VAX VMS. IDE's technical support is the benchmark by which I measure other vendors.

**Vendor Comments:**

## SEE Product Critique

**Product Name:** Virtual Software Factory (VSF)  
**Vendor Name:** VSF  
**Special Operating Environment:**

**Version Number:** 3.8  
**Hardware Platform:** Sun  
**Operating System:** Unix

**Project Description:** Client/server application requirements gathering and code generation.

**Years of software experience:** 15  
**Years of experience with product:** 2  
**Last time product was used:** Mar 1993

**Overall Impression of this Product:**  
Good  
**Quality of Vendor Support:**  
Good

**Notable Strengths of this Product:**  
Integration of graphics and text documents.

**Notable Weaknesses of this Product:**  
Poor interface (user interface is not user definable).

**Advice for Potential Users or Buyers of this Product:**  
Start with a few licenses and build "core" experience.

**Vendor Comments:**

## **Appendix F: SEE Product Features Matrix**

Pages F-3 through F-6 contain the portion of the STSC SEE Product Vendor Questionnaire Form that was used to collect the information that is shown in the SEE Product Features Matrix, which begins on page F-7. The form is included to assist the reader in understanding the information provided in the matrix.

STSC SEE Product Vendor Questionnaire Form

**A. Category**

For STSC purposes, Software Engineering Environment (SEE) products are divided into two categories, which are listed and defined below. The STSC uses the NIST/ECMA Reference Model for SEE Frameworks as its guide for SEE framework services.

- A **SEE Framework** is a product providing services that facilitate and support tools integration and interoperability, including external (third-party) tools; and data sharing by a repository.
- A **Populated SEE Framework** is a SEE Framework that has integrated within it end-user tools that support one or more phases of the software life cycle.

Is your product...

A SEE Framework? ☐, A Populated SEE Framework? ☐,  
Other ☐ (describe) \_\_\_\_\_

**B. Standards Supported**

1. Which platform/operating system standards does your product currently support?

POSIX ☐, Unix ☐,  
Other ☐ (list) \_\_\_\_\_

2. Which framework standards does your product currently support?

PCTE ☐, ATIS ☐, CAIS-A ☐, CORBA ☐, BMS ☐, Tooltalk ☐,  
Other ☐ (list) \_\_\_\_\_

3. Which graphical user interface standards does your product currently support?

X Windows ☐, Openlook ☐, MOTIF ☐, PHIGS ☐, GKS ☐,  
Other ☐ (list) \_\_\_\_\_

4. Which data and repository standards does your product currently support?

IRDS ☐, CDIF ☐, ASIS ☐, PDES ☐, STEP ☐,  
Other ☐ (list) \_\_\_\_\_

**C. Tool Integration**

1. Please indicate in the following table which types of tool integration are used by your product.

Type of Tool Integration	Description	? Yes/No
Platform	Operating system and network are transparent to the tools.	
Presentation	User interface for all tools is consistent and has same appearance to the user	
Data	Data is transferred between tools either directly or by a shared repository.	
Control	Communication between tools is by triggers, daemons or messaging facilities.	
Other (Describe)		

2. In the table below please list the external tools that have been successfully integrated into your product. Also, for each external tool, please indicate its function (project management, CM, documentation, metrics, process, software development, etc.), whether it is included in the purchase price of your product, whether it is available for distribution with your product, and whether it requires a special module to run with your product.

External Tool	Function	Included in SEE Price? (Yes/No)	Available for distri- bution? (Yes/No)	Special Module Required? (Yes/No)

3. Does integration of an external (third party) tool into your product require encapsulation?  
Yes ☐, No ☐

**D. Repository**

- Does your product include a repository of software development objects? Yes ☐, No ☐
- Can external (third party) tools easily share data with your repository? Yes ☐, No ☐
- Which of the following structures best describes your repository?
  - ☐ Inclusive, predefined
  - ☐ Extendible
  - ☐ Modifiable
  - ☐ Other (describe) \_\_\_\_\_
- Which items in the list below can be stored and shared in the repository of your product without modification of the product by the user?  
 Requirements Information ☐, Design Information ☐, Code ☐,  
 Project Schedule ☐, Process Artifacts ☐, Process Roles ☐, Metrics ☐,  
 Other ☐ (list) \_\_\_\_\_
- Can the repository structure be easily modified by a trained SEE implementor? Yes ☐, No ☐

**E. Software Life-cycle Support**

1. Which of the software life cycle phases listed in the following table does your product support? Please indicate support by providing the third party tool name or describing the internal SEE function.

Life Cycle Phase	Third Party Tool Name	Internal SEE Function (Describe)
a. Requirements Definition		
b. Design		
c. Coding - Ada		
d. Coding - Other		
e. Debugging		
f. Testing		
g. Target System Installation		
h. Post-Development Support/Maintenance		

2. Which of the software development tasks listed in the table below does your product support? Please indicate support by providing the third party tool name or describing the internal SEE function.

Software Development Task	Third Party Tool Name	Internal SEE Function (Describe)
a. Documentation Preparation		
b. Requirements Tracing		
c. Problem Report Management		
d. Quality Assurance		
e. Software Reuse Management		
f. Process Definition		
g. Process Enactment		
h. Configuration Management		
i. Change Management		
j. Metrics		
k. Software Reverse Engineering		
l. Software Re-engineering		

**F. Process and Metrics Support**

1. Does your product implement a standard process model? No ☐  
Yes ☐ Which one? \_\_\_\_\_
2. What flexibility does a SEE implementor/manager have in selecting and defining the features of the development process? \_\_\_\_\_
3. What is the average number of hours companies have expended to implement their process models on your product?  
Less than 200 ☐ Between 200 and 1000 ☐ More than 1000 ☐ Not Applicable ☐
4. How does your product support user roles, tasks, and assignments? \_\_\_\_\_
5. How is team communication supported by your product? \_\_\_\_\_
6. How is the collection and analysis of metrics supported by your product? \_\_\_\_\_
7. Which metrics are collected automatically? \_\_\_\_\_

**G. Project Management Support**

Which project management areas listed in the table below does your product support? Please indicate support by providing the third party tool name or describing the internal SEE function.

Project Management Area	Third Party Tool Name	Internal SEE Function (Describe)
a. Software Size Estimation		
b. Project Cost Estimation		
c. Scheduling		
d. Resource Management		
e. Task Tracking		
f. Risk Analysis		
g. Metrics		
h. Project History		

**H. User Interface**

1. Which of the following best describe the user interface of your product (check all that apply)?  
 GUI ☐, Textual ☐, Windows ☐, X Windows ☐, Motif ☐, Menu Driven ☐,  
 Other ☐ (describe) \_\_\_\_\_
2. Can the user or SEE implementor/manager easily customize the interface? Yes ☐, No ☐.

**I. Scalability**

Please indicate the largest project, in source lines of code (SLOC) or number of software engineers or both, that has been successfully developed or is being developed using your product.

SLOC: Less than 50,000 ☐; Between 50,000 and 1,000,000 ☐; More than 1,000,000 ☐  
 Number of Software Engineers: Less than 50 ☐, Between 50 and 300 ☐, More than 300 ☐

**J. Security**

Describe how your product provides security for the development process and the data objects used.

\_\_\_\_\_

\_\_\_\_\_

**K. Customer Services**

Please indicate which of the services listed below are provided with your product to support users. Also, please indicate for each service provided whether there is an additional cost that is not included in the purchase price.

Installation ☐ Add'l. cost ☐, Training ☐ Add'l. cost ☐,  
 Consulting ☐ Add'l. cost ☐, On-line Help ☐ Add'l. cost ☐,  
 Technical Hot-line ☐ Add'l. cost ☐, Customization assistance ☐ Add'l. cost ☐,  
 Tool Integration assistance ☐ Add'l. cost ☐,  
 Other ☐ (describe) \_\_\_\_\_  
 Add'l. cost ☐

(This page is intentionally blank.)

*Software Technology Support Center*

SEE Product (version)	AISLE Tools (2.0)	AIX SDE Workbench/6000 Integrator/6000 (1.2)	Apex (1.2)	C - Development Environment S/W Through Pictures (1.1)
<b>A. Category</b>	Integrated toolset	Populated SEE Framework	Populated SEE Framework	Expanded CASE Tool or PSE
<b>B. Standards Supported</b> Platform/OS Framework GUI  Data/repository	POSIX, Unix  X Windows	POSIX, TCP/IP BMS X Windows, Motif  SQL/RDBMS	POSIX, Unix BMS X Windows, Motif  ASIS	BMS X Windows, OpenLook, Motif
<b>C. Tool Integration</b> Type Integrated external tools Requires encapsulation?	Platform, Presentation, Data Teamwork No		Platform, Presentation, Data, Control Yes - several No	
<b>D. Repository</b> Includes S/W dev. objects? External tool data sharing? Structure Stores & shares:      Structure easily modified?	No No Internal Code Metrics    No		Yes Yes Extendible Req's. information Design information Code Project schedule Process artifacts Process roles Metrics Yes	
<b>E. S/W Life Cycle Support</b> 1. Life Cycle Phase: Req's. definition Design Coding - Ada Coding - other Debugging Testing Target system installation Maintenance	ADADL ADADL, QUALGEN CDADL  TESTGEN  GRAFBROWSE	Teamwork Rational  Program Debugger  CMVC/6000	Teamwork, Rose Teamwork, Rose Apex  Apex TestMate All Ada compilers (All of above)	STRAND-RT   "T" from PEI
2. S/W Development Task: Documentation prep Req's. tracing Problem report mgt. Quality assurance S/W reuse mgt. Process definition Process enactment Configuration mgt. Change mgt. Metrics S/W reverse engineering S/W re-engineering	DOCGEN  QUALGEN ADADL  QUALGEN REV Ada REV Ada	Interleaf, FrameMaker  CMVC/6000  Process Weaver Process Weaver	SODA SODA  Apex, TestMate Apex Apex Apex Apex AdaAnalyzer, AdaQuest Rose/Ada Apex	      STRAND-RT STRAND-RT
<b>F. Process &amp; Metrics Support</b> Std. process model Flexibility Avg. hours Support roles, etc. Team communication Metrics collection Which metrics	No Use what you want >1000  Std. formats, reviews Automatic >250 code metrics	No	Yes - Rational approach Completely 200-1000  Integrated CM User-programmable User-programmable	No Flexible  Multiuser project DB Can collect

*Appendix F: SEE Product Features Matrix*

SEE Product (version)	AISLE Tools (2.0)	AIX SDE Workbench/6000 Integrator/6000 (1.2)	Apex (1.2)	C - Development Environment S/W Through Pictures (1.1)
<b>G. Project Mgt. Support</b> S/W size estimation Project cost estimation Scheduling Resource mgt. Task tracking Risk analysis Metrics Project history	QUALGEN		AdaQuest, AdaAnalyzer CMVC	
<b>H. User Interface</b>  Easily customized?	GUI, X Windows, Motif, Textual, Menu- Driven No	Motif	GUI, X Windows, Motif, Menu Driven Yes	X Windows, Motif, Menu-driven, O-O
<b>I. Scalability</b> SLOC Number of S/W engineers	50K-1M	50K-1M 50-300	>1M >300	50K-1M <50
<b>K. Customer Services</b> Installation Training Consulting On-line help Technical hot line Customization assistance Tool integration assistance	Add'l. cost Add'l. cost  No add'l. cost		Add'l. cost Add'l. cost Add'l. cost No add'l. cost Add'l. cost Add'l. cost Add'l. cost	

*Software Technology Support Center*

SEE Product (version)	CodeCenter ObjectCenter (4.0)	COHESION ASD/SEE (2.2)	COHESION Team/SEE (1.0)	COHESIONworX (2.0)
<b>A. Category</b>	Populated SEE Framework	Populated SEE Framework	Populated SEE Framework	Populated SEE Framework
<b>B. Standards Supported</b> Platform/OS Framework GUI  Data/repository	Solaris, HP-UX, (others) BMS X Windows, OpenLook, Motif	POSIX CORBA X Windows, Motif, PHIGS, GKS CDIF	POSIX, Unix, DCE CORBA, MCMS, X3H6 Motif  ORACLE-based	POSIX, Unix, DCE CORBA, MCMS, X3H6 OpenLook, Motif
<b>C. Tool Integration</b> Type Integrated external tools Requires encapsulation?			Platform, Presentation, Data, Control (various) Yes	Platform, Presentation, Data, Control (various) Yes
<b>D. Repository</b> Includes S/W dev. objects? External tool data sharing? Structure Stores & shares:       Structure easily modified?				
<b>E. S/W Life Cycle Support</b> 1. Life Cycle Phase: Req's. definition Design Coding - Ada Coding - other Debugging Testing Target system installation Maintenance	C++ ObjectCenter  CodeCenter/ObjectCtr.  CodeCenter/ObjectCtr.	RTM Teamwork VAX Ada C, C++, (others)  DECset, DECTest Mgr.  RTM, PCMS, DECplan	RTM Teamwork, STP, OMT DEC Ada, Apex DEC FUSE dbx, LaDEBUG	DEC FUSE DEC FUSE dbx, LaDEBUG DEC FUSE Profiles
2. S/W Development Task: Documentation prep Req's. tracing Problem report mgt. Quality assurance Reuse mgt. Process definition Process enactment Configuration mgt. Change mgt. Metrics S/W reverse engineering S/W re-engineering	(Yes) (Yes)	Interleaf, PCMS RTM PCMS PCMS, (others) (next version) PCMS, DECplan (next version)	Interleaf RTM PCMS  PCMS Process Assistant™ PCMS PCMS Amadeus	Interleaf RTM PCMS  PCMS PCMS PCMS Amadeus
<b>F. Process &amp; Metrics Support</b> Std. process model Flexibility Time, hours Support roles, etc. Team communication Metrics collection Which metrics	No  Programmers tool Browses Performance analysis	Yes - DOD-STD-2167A Highly flexible  Tailored by user Mail, conference notes Ada code counter	Yes Complete control <200 Control Plan Mail Automated collection Many raw code metrics	No Process not imposed <200 Third party Mail

*Appendix F: SEE Product Features Matrix*

SEE Product (version) Company	CodeCenter ObjectCenter (4.0)	COHESION ASD/SEE (2.2)	COHESION Team/SEE (1.0)	COHESIONworX (2.0)
<b>G. Project Mgt. Support</b> S/W size estimation Project cost estimation Scheduling Resource mgt. Task tracking Risk analysis Metrics Project history		DECplan DECplan DECplan DECplan DECplan  DECplan	(Next version)	
<b>H. User Interface</b>  Easily customized?	GUI, Textual, X Windows, Motif, O-O	GUI, Textual, Windows, X Windows, Motif, Menu-driven, O-O	GUI, X Windows Motif, EXLdesktop  Yes	GUI, Motif, icons, drag- and-drop, pull-down menus Yes
<b>I. Scalability</b> SLOC Number of S/W engineers	>1M >300	>1M >300	>1M 50-300	50K-1M 50-300
<b>K. Customer Services</b> Installation Training Consulting On-line help Technical hot line Customization assistance Tool integration assistance	Add'l. cost No add'l. cost		Add'l. cost Add'l. cost Add'l. cost  No add'l. cost Add'l. cost Add'l. cost	Add'l. cost Add'l. cost Add'l. cost No add'l. cost No add'l. cost Add'l. cost Add'l. cost

*Software Technology Support Center*

SEE Product (version)	Cradle SEE (1.8)	EAST Environment	Emeraude V12 (V12.4)	FreedomWorks (1.3)
<b>A. Category</b>	Populated SEE Framework	Environment	Framework	SEE Framework
<b>B. Standards Supported</b> Platform/OS Framework GUI  Data/repository	POSIX, Unix, VMS BMS X Windows, OpenLook, Motif, OpenDialog CDIF, ASCII	Sun, IBM PCTE X Windows, Motif	POSIX PCTE X Windows, OpenLook, Motif PCTE	Unix PCTE X Windows, OpenLook, Motif
<b>C. Tool Integration</b> Type Integrated external tools Requires encapsulation?	Platform, Presentation, Data, Control Interleaf, FrameMaker No	(Some)	(Some)	Platform, Presentation, Data (Many) Yes
<b>D. Repository</b> Includes S/W dev. objects? External tool data sharing? Structure Stores & shares:   Structure easily modified?	Yes Yes Inclusive Req's. information Design information Code Project schedules Process artifacts Process roles Metrics Yes	Flexible       Yes	Flexible, Basic       Yes	Yes Yes Extendible, modifiable Req's. information Design information Code Project schedule Process artifacts Process roles Metrics Documentation Yes
<b>E. S/W Life Cycle Support</b> 1. Life Cycle Phase: Req's. definition Design Coding - Ada Coding - other Debugging Testing Target system installation Maintenance	Cradle SEE Analysis Cradle SEE Design Cradle SEE Design Cradle SEE Design Cradle Animation/PERF Cradle Animation/PERF  Cradle SEE CM	ObjectMaker, Teamwork, STP Teamwork, STP AdaNice, Alsys CNice, Logiscope  Logiscope	Native OS compiler	STP, Teamwork Alsys, Verdix C, C++, Fortran (various) Logiscope  Object CM
2. S/W Development Task: Documentation prep Req's. tracing Problem report mgt. Quality assurance S/W reuse mgt. Process definition Process enactment Configuration mgt. Change mgt. Metrics S/W reverse engineering S/W re-engineering	Document Composer Cradle SEE Cradle CM Cradle CM Cradle SEE Analysis & Design Animation & PERF Cradle CM Cradle CM Cradle Cradle Reverse Eng. Cradle Reverse Eng.	(Built-in function)   (Built-in function) (Built-in function) (Built-in function)		Freedom/Documeter  Object CM  Freedom/Object Reuse Object CM Object CM Object CM Object CM Logiscope STP, Teamwork
<b>F. Process &amp; Metrics Support</b> Std. process model Flexibility Avg. hours Support roles, etc. Team communication Metrics collection Which metrics	Yes - Yourdon Process is extendible. N/A User Profile config. tool Cradle CM Reporting tools User & data activities	Yes Full customization  Built-in Built-in Built-in	No Open to integration  Not provided Not provided Not provided	Yes - DOD-STD-2167A Very configurable <200 Fully implemented Various tools Logiscope

*Appendix F: SEE Product Features Matrix*

SEE Product (version)	Cradle SEE (1.8)	EAST Environment	Emeraude V12 (V12.4)	FreedomWorks (1.3)
<b>G. Project Mgt. Support</b> S/W size estimation Project cost estimation Scheduling Resource mgt. Task tracking Risk analysis Metrics Project history	(Limited) (Limited) (Future release) Cradle's User Profile Cradle CM (limited) Cradle PERF Various tools Cradle CM	(Built-in functions)		Freedom/Project Mgr. Freedom/Project Mgr. Freedom/Project Mgr. Freedom/Project Mgr. Freedom/Project Mgr. Artemis Artemis Artemis
<b>H. User Interface</b>  Easily customized?	GUI, Textual, X Windows, Motif  No	GUI, Textual, Motif, Menu-driven, O-O	GUI, Textual, Windows, Motif, Menu-driven	Textual, Motif, OpenLook Yes
<b>I. Scalability</b> SLOC Number of S/W engineers	50K-1M <50	50K-1M 50-300	>1M <50	>1M >300
<b>K. Customer Services</b> Installation Training Consulting On-line help Technical hot line Customization assistance Tool integration assistance	No add'l. cost Add'l. cost Add'l. cost No add'l. cost No add'l. cost Add'l. cost Add'l. cost			Add'l. cost Add'l. cost Add'l. cost Add'l. cost Add'l. cost Add'l. cost Add'l. cost

Software Technology Support Center

SEE Product (version)	Information Engineering Facility (5.1)	Integrated System Development (1.0)	IPSYS ToolBuilder (1.3)	LIFE*CYCLE (2.2)
<b>A. Category</b>	Populated Framework	SEE Framework SEE Builder	SEE Builder	Populated SEE Framework
<b>B. Standards Supported</b> Platform/OS Framework GUI Data/repository	POSIX  X Windows, OpenLook, Motif, X/11	POSIX PCTE, BMS X Windows, Motif, Windows CDIF, Software One	XPG III, POSIX IEEE PCTE X Windows, OpenLook, Motif	Unix ATIS X Windows, OpenLook, Motif ATIS
<b>C. Tool Integration</b> Type Integrated external tools Requires encapsulation?				Platform, Presentation, Data, Control  Yes
<b>D. Repository</b> Includes S/W dev. objects? External tool data sharing? Structure Stores & shares:    Structure easily modified?	Limited Extensibility	C w/cur. ver. A&B w/v2	Flexible	Yes Yes Extendible Req's. information Design information Code Process roles  Yes
<b>E. S/W Life Cycle Support</b> 1. Life Cycle Phase: Req's. definition Design Coding - Ada Coding - other Debugging Testing Target system installation Maintenance		(Contact vendor)		Verilog ASA, RD-100 StP EOSD, Life*Design VADSPRO, Life*Ada  Verdix
2. S/W Development Task: Documentation prep Req's. tracing Problem report mgt. Quality assurance S/W reuse mgt. Process definition Process enactment Configuration mgt. Change mgt. Metrics S/W reverse engineering S/W re-engineering	JMA Appl. Templates	Interleaf  ACTBAT PVCs Process Weaver Process Weaver		Interleaf Life*LINK Life*FLOW  Life*FLOW Life*FLOW (intrinsic) (intrinsic)
<b>F. Process &amp; Metrics Support</b> Std. process model Flexibility Avg. hours Support roles, etc. Team communication Metrics collection Which metrics	Yes Yes  Multilevels of access Yes Yes	Customizable Full flexibility  N/A Process Weaver Task mgt.		ESA, DOD-STD-2167A Fair 200-1000 Built-in Shared access Data types History

*Appendix F: SEE Product Features Matrix*

SEE Product (version)	Information Engineering Facility (5.1)	Integrated System Development (1.0)	IPSYS ToolBuilder (1.3)	LIFE*CYCLE (2.2)
<b>G. Project Mgt. Support</b> S/W size estimation Project cost estimation Scheduling Resource mgt. Task tracking Risk analysis Metrics Project history	IEF Project Manager IEF Project Manager IEF Project Manager IEF Project Manager IEF Project Manager  IEF Project Manager			(intrinsic)
<b>H. User Interface</b>  Easily customized?	GUI, Windows, X Windows, Motif	Windows, X Windows, Motif	GUI, Textual, Motif, X Windows, Menu-driven, O-O	GUI, X Windows, Motif, Menu-driven Yes
<b>I. Scalability</b> SLOC Number of S/W engineers	>1M >300	>1M >300	>1M >300	50K-1M <50
<b>K. Customer Services</b> Installation Training Consulting On-line help Technical hot line Customization assistance Tool integration assistance				Add'l. cost Add'l. cost Add'l. cost  Add'l. cost Add'l. cost

*Software Technology Support Center*

SEE Product (version)	Metatool Specification Driver (2.0)	METHODMANAGER (2.7.0)	Objectime (4.0)	ObjectMaker (2.1)
<b>A. Category</b>	SEE Framework	Populated SEE Framework	Expanded CASE Tool or PSE	Expanded CASE Tool or PSE
<b>B. Standards Supported</b> Platform/OS Framework GUI  Data/repository	Unix  Character-based I/F	IBM MVS, VM/CMS BMS MS Windows	SUN OS, HP-UX  X Windowe, OpenLook	Unix, IBM, Sun, HP PCTE, OLE, DDE X Windows, Motif, MS Windows
<b>C. Tool Integration</b> Type Integrated external tools Requires encapsulation?	Platform, Presentation, Data, Control (various) No	(various)	(various)	(various)
<b>D. Repository</b> Includes S/W dev. objects? External tool data sharing? Structure Stores & shares:   Structure easily modified?	No No Code   No	Flexible	SCCS/RCS	Flexible, basic
<b>E. S/W Life Cycle Support</b> 1. Life Cycle Phase: Req's. definition Design Coding - Ada Coding - other Debugging Testing Target system installation Maintenance	(internal) (internal)  (internal)	ADW Bachman, IEF	GNU C++, C Front	Yes Yes Yes C, C++  Yes Yes Yes
2. S/W Development Task: Documentation prep Req's. tracing Problem report mgt. Quality assurance S/W reuse mgt. Process definition Process enactment Configuration mgt. Change mgt. Metrics S/W reverse engineering S/W re-engineering			FrameMaker	(Many) Yes Yes  Planned/RAPID Yes
<b>F. Process &amp; Metrics Support</b> Std. process model Flexibility Avg. hours Support roles, etc. Team communication Metrics collection Which metrics	Yes Broad range 200-1000	No Fully tailorable  May be defined Use of common objects	No Flexible  Graphical, Executable	Yes Build and tailor own  Yes Yes Yes

*Appendix F: SEE Product Features Matrix*

SEE Product (version)	Metatool Specification Driver (2.0)	METHODMANAGER (2.7.0)	Objectime (4.0)	ObjectMaker (2.1)
<b>G. Project Mgt. Support</b> S/W size estimation Project cost estimation Scheduling Resource mgt. Task tracking Risk analysis Metrics Project history				
<b>H. User Interface</b>  Easily customized?	Textual	Windows, Menu-driven	GUI, Windows, Menu-driven, O-O	GUI, Windows, Motif, X Windows, Menu-driven, O-O
<b>I. Scalability</b> SLOC Number of S/W engineers	50K-1M <50		<50	50K-1M 50-300
<b>K. Customer Services</b> Installation Training Consulting On-line help Technical hot line Customization assistance Tool integration assistance	Add'l. cost			

*Software Technology Support Center*

SEE Product (version)	Objectory (3.3)	OOSD/Ada - Software Through Pictures (1.2)	PCTE Workbench (1.1)	Process Weaver (2.0)
<b>A. Category</b>		Expanded CASE Tool or PSE	Populated SEE Framework	SEE Framework
<b>B. Standards Supported</b> Platform/OS Framework GUI Data/repository	BMS '93 OpenLook, Motif  CDIF	SUN, HP, DEC, IBM BMS X Windows, OpenLook, Motif DB schema	Unix PCTE, Tooltalk X Windows, Motif  PCTE	Unix, Windows NT PCTE, BMS X Windows, Motif
<b>C. Tool Integration</b> Type Integrated external tools Requires encapsulation?		(various)	Platform, Presentation, Data, Control FrameMaker, Epoch Yes	Platform, Presentation, Data, Control (75 tools) No
<b>D. Repository</b> Includes S/W dev. objects? External tool data sharing? Structure Stores & shares:   Structure easily modified?	Basic	Flexible	Yes Yes Modifiable Req's. information Design information Code Project schedule Process artifacts Process roles Metrics	Yes Yes Modifiable Process artifacts Process roles Metrics   Yes
<b>E. S/W Life Cycle Support</b> 1. Life Cycle Phase: Req's. definition Design Coding - Ada Coding - other Debugging Testing Target system installation Maintenance		STRAND-RT	FrameMaker FrameMaker Epoch Epoch	Yes Yes Yes Yes Yes Yes Yes
2. S/W Development Task: Documentation prep Req's. tracing Problem report mgt. Quality assurance S/W reuse mgt. Process definition Process enactment Configuration mgt. Change mgt. Metrics S/W reverse engineering S/W re-engineering		STRAND-RT STRAND-RT	FrameMaker Hypertext linking  Scripting Language	Yes Yes Yes Yes Yes Yes Yes Yes Yes Yes Yes
<b>F. Process &amp; Metrics Support</b> Std. process model Flexibility Avg. hours Support roles, etc. Team communication Metrics collection Which metrics	Yes Development CASE's  CASE Modeling Multiuser support Defined dev. process	No  Multiuser project DB Can collect	No Entirely customizable	No Copy & modify <200 Defined & assigned Delegation by Petri nets Activity Incident Mgr. Task durations, error loops

*Appendix F: SEE Product Features Matrix*

SEE Product (version)	Objectory (3.3)	OOSD/Ada - Software Through Pictures (1.2)	PCTE Workbench (1.1)	Process Weaver (2.0)
<b>G. Project Mgt. Support</b> S/W size estimation Project cost estimation Scheduling Resource mgt. Task tracking Risk analysis Metrics Project history				(Interface with PM tools)
<b>H. User Interface</b>  Easily customized?	GUI, X Windows, Motif, O-O	X Windows, Motif, Menu-driven	X Windows, Motif, Menu-driven  Yes	GUI, Motif  Yes
<b>I. Scalability</b> SLOC Number of S/W engineers	50K-1M 50-300	>1M 50-300	<50	50-300
<b>K. Customer Services</b> Installation Training Consulting On-line help Technical hot line Customization assistance Tool integration assistance			Add'l. cost Add'l. cost Add'l. cost  No add'l. cost Add'l. cost Add'l. cost	No add'l. cost Add'l. cost Add'l. cost  No add'l. cost Add'l. cost Add'l. cost

*Software Technology Support Center*

SEE Product (version)	PRO-IV Workbench (1.0)	ProSLCSE (2.2)	SoftBench, SoftBench/C++ (3.3)	Software Backplane (3.0.1)
<b>A. Category</b>		Populated SEE Framework	Populated SEE Framework	SEE Framework
<b>B. Standards Supported</b> Platform/OS Framework GUI  Data/repository		Unix PCTE, ToolTalk X Windows, OpenLook, Motif PCTE	POSIX BMS X Windows, OpenLook, Motif	ATIS, BMS X Windows, OpenLook, Motif, Command Line IRDS, CDIF
<b>C. Tool Integration</b> Type Integrated external tools Requires encapsulation?		(various)	(various)	(various)
<b>D. Repository</b> Includes S/W dev. objects? External tool data sharing? Structure Stores & shares:    Structure easily modified?	Req's. information Design information Code Project schedule Process artifacts Process roles Metrics	Flexiable		Flexible
<b>E. S/W Life Cycle Support</b> 1. Life Cycle Phase: Req's. definition Design Coding - Ada Coding - other Debugging Testing Target system installation Maintenance		CADRE VADSpro Jovial (ACT)  SEE	Artifex, Foresight	CADRE/IDE CADRE/IDE  VERDIX, Centerline  (Integrated by user) (Integrated by user) (Integrated by user)
2. S/W Development Task: Documentation prep Req's. tracing Problem report mgt. Quality assurance S/W reuse mgt. Process definition Process enactment Configuration mgt. Change mgt. Metrics S/W reverse engineering S/W re-engineering		Interleaf	Interleaf, FrameMaker  Qualtrak, (others) Logiscope, (others) Cadre/HP, (others) HP-Synervision	Interleaf (As integrated)  (As integrated)
<b>F. Process &amp; Metrics Support</b> Std. process model Flexibility Avg. hours Support roles, etc. Team communication Metrics collection Which metrics		No Definable process  Infrastructure E-mail, artifacts Process enactment	No Completely  Textually, Graphically BMS auto triggers Automatic collection	No Completely  Access control Perm. Using access controls Definable

*Appendix F: SEE Product Features Matrix*

SEE Product (version)	PRO-IV Workbench (1.0)	ProSLCSE (2.2)	SoftBench, SoftBench/C++ (3.3)	Software Backplane (3.0.1)
<b>G. Project Mgt. Support</b> S/W size estimation Project cost estimation Scheduling Resource mgt. Task tracking Risk analysis Metrics Project history			Teamwork S/W Teamwork S/W Auto Plan Auto Plan HP-Synervision  HP-Synervision	
<b>H. User Interface</b>  Easily customized?		GUI, X Windows, Motif, OpenLook	GUI, X Windows, Motif, Menu-driven, O-O	Command line
<b>I. Scalability</b> SLOC Number of S/W engineers		<200K <50	>1M >300	>1M 50-300
<b>K. Customer Services</b> Installation Training Consulting On-line help Technical hot line Customization assistance Tool integration assistance				

SEE Product (version)	STRAND-RT - Software Through Pictures (4.2D)	VSF, MWB-SF/AWB-SF (3.8)
<b>A. Category</b>	Expanded CASE tool or PSE	SEE Builder
<b>B. Standards Supported</b> Platform/OS Framework GUI  Data/repository	SUN, (others) BMS, (others) X Windows, OpenLook, Motif CDIF, DB Schema	POSIX, OS/2, VMS PCTE, BMS X Windows, Motif  CDIF, IDEF
<b>C. Tool Integration</b> Type Integrated external tools Requires encapsulation?	(various)	
<b>D. Repository</b> Includes S/W dev. objects? External tool data sharing? Structure Stores & shares:         Structure easily modified?	Flexible	Semantic Network
<b>E. S/W Life Cycle Support</b> 1. Life Cycle Phase: Req's. definition Design Coding - Ada Coding - other Debugging Testing Target system installation Maintenance	"T" from PEI	
2. S/W Development Task: Documentation prep Req's. tracing Problem report mgt. Quality assurance S/W reuse mgt. Process definition Process enactment Configuration mgt. Change mgt. Metrics S/W reverse engineering S/W re-engineering		
<b>F. Process &amp; Metrics Support</b> Std. process model Flexibility Avg. hours Support roles, etc. Team communication Metrics collection Which metrics	No Not prescriptive   Multiuser project DB Can collect	Yes Total  Depends on method Export/import

*Appendix F: SEE Product Features Matrix*

SEE Product (version)	STRAND-RT - Software Through Pictures (4.2D)	VSF, MWB-SF/AWB-SF (3.8)
<b>G. Project Mgt. Support</b> S/W size estimation Project cost estimation Scheduling Resource mgt. Task tracking Risk analysis Metrics Project history		
<b>H. User Interface</b>  Easily customized?	X Windows, Motif, Menu-driven, O-O	GUI, X Windows, Motif, O-O
<b>I. Scalability</b> SLOC Number of S/W engineers	>1M 50-300	>1M <50
<b>K. Customer Services</b> Installation Training Consulting On-line help Technical hot line Customization assistance Tool integration assistance		

(This page is intentionally blank.)

## **Appendix G: Standards**

## **G.1 Standards**

### **G.1.1 Framework Standards**

#### **G.1.1.1 Portable Common Tool Environment (PCTE)**

PCTE is "a standard for a public tool interface (PTI) for an open repository" [WAKEMAN 93]. The interface supports the use of tools and is designed to allow program portability by providing machine-independent access to the framework services. PCTE was accepted as an ECMA standard in 1990. An abstract specification of the standard, which is independent of any programming language, is ECMA-149. The mapping of the abstract operations to operations in a programming language are referred to as a *binding*. The language binding for Ada is ECMA-162, and C is ECMA-158.

#### **G.1.1.2 A Tools Integration Standard (ATIS)/CASE Integration Services (CIS)**

ATIS is a draft standard that uses an object-oriented approach to tools integration [ROCK-EVANS 93a]. ATIS grew out of a joint effort between Atherton Technology and Digital Equipment Corporation to standardize Integrated Project Support Environment (IPSE) tool integration services [SHU 92]. CIS is the name sometimes given to the version of ATIS, issued by a CIS working group, based on both the original ATIS and a merger of ATIS with the ANSI Information Resource Dictionary System (IRDS) [KARRER 90].

#### **G.1.1.3 Ada Programming Support Environment (APSE)/Common APSE Interface Set (CAIS-A)**

CAIS-A is a DoD standard (DOD-STD-1838A) for tool portability and interoperability in the development of APSEs. It is sponsored by the DoD's Ada Joint Program Office (AJPO) [ROCK-EVANS 93a].

#### **G.1.1.4 Portable Common Interface Set (PCIS)**

PCIS is a framework environment definition based on the ECMA PCTE. The PCIS program is an international effort initiated by a NATO special working group.

#### **G.1.1.5 Object Management Architecture (OMA)/Common Object Request Broker Architecture (CORBA)**

OMA is a specification of object-oriented services that provide a common framework for distributed applications development. OMA covers two models: a RM and an abstract

object model. The Object Request Broker (ORB) is a component of the RM; it facilitates central communication and integration between independently developed applications. CORBA is the specification for the ORB being distributed by the Object Management Group (OMG) and X/Open [ROCK-EVANS 93a].

## **G.1.2 Graphical User Interface (GUI) Standards**

With the possible exception of Common Desktop Environment (CDE), the following GUI standards are de facto. The degree to which they are standard depends on how close to true look and feel the standard is.

### **G.1.2.1 Motif**

Motif is a product from the Open Software Foundation (OSF). It is similar to Microsoft's Windows, and it is designed for use with a UNIX operating system [ROCK-EVANS 93a].

### **G.1.2.2 X-Windows**

X-Windows is a "networked windowing system using a client-server model" [ROCK-EVANS 93a].

### **G.1.2.3 Microsoft Windows 3.x**

Microsoft's Windows is a well-known, proprietary user interface that is virtually a de facto standard for PCs.

### **G.1.2.5 Common Desktop Environment (CDE)**

CDE is a common UNIX standard and tool, developed by the Common Open Software Environment (COSE) group, that allows software developers to work with a single set of programming interfaces [LAWTON 93].

## **G.1.3 Data Repository Standards**

### **G.1.3.1 Structured Query Language (SQL)**

SQL is a DoD-approved language standard that serves as a data definition language (DDL) and data manipulation language (DML) for relational database management systems (DBMSs). Current work by the ANSI Technical Committee X3H2 is focusing on extending the language, including object models.

### **G.1.3.2 ISO Information Resource Dictionary System (IRDS)**

ISO IRDS is a family of standards focused on data integration and data management. Standards completed to date are an IRDS framework (a repository architecture) and an IRDS services interface. SQL is used to define the standards [ROCK-EVANS 93a].

## **G.1.4 Platform Standards**

Platform standards provide portability of a SEE across operating systems.

### **G.1.4.1 Portable Operating System Interface (POSIX)**

POSIX (pronounced "pahz-icks") is a suite of platform standards being developed by the IEEE/Computer Society, specifically the Technical Committee on Operating Systems. The POSIX interface is derived from the UNIX operating system, but is not a standard for UNIX. Refer to [BRASHARES 93] for a comprehensive description of POSIX.

### **G.1.4.2 UNIX**

UNIX and its derivatives are the quintessential commodity operating systems—operating systems derived from the same root and sold by many vendors. Basically, UNIX developments are coming from two consortia: UNIX International (UI) and OSF. Efforts are under way by UI to make UNIX System V version 4 (SVR4) available on PCs.

## **G.1.5 Tool Interchange Standards**

Tool interchange standards govern the passing of data between tools [ROCK-EVANS 93a].

### **G.1.5.1 CASE Data Interchange Format (CDIF)**

CDIF is a set of interim standards, issued by the Electronic Industries Association (EIA), for transferring information between CASE tools. The purpose of the CDIF is to eliminate the need for custom interfaces between tools and between tools and repositories [NGCR 93a].

### **G.1.5.2 P1175**

The IEEE Standard P1175 is a RM for integrating computer-aided engineering tools, computer-aided system engineering tools, and CASE tools into a family.

### **G.1.5.3 Product Data Exchange Specification (PDES)/Initial Graphics Exchange Specification (IGES).**

PDES/IGES are general engineering product standards that describe objects within the scope of Computer-Aided Design/Computer-Aided Manufacture (CAD/CAM) systems. They represent inputs by the United States to the ISO Standard for the Exchange of Product Model Data (STEP).

### **G.1.6 Security**

The policy enforcement services of a SEE framework provide adequate security for most software systems. Exceptional security needs can arise with SEEs used within the DoD and military services and with other government agencies such as the National Security Agency. This report does not address those exceptional needs, although the STSC's SEE Technology Domain team is working with agencies concerned about those needs.

## **G.2 Organizations Developing and Promoting SEE Standards**

The following organizations are standards-making bodies or are involved in standards-making activities such as promoting specific standards, preparing candidates for standards, or coordinating standards activities.

### **G.2.1 The National Institute for Standards and Technology (NIST)**

The NIST, formerly the National Bureau of Standards, is a major agency in the U. S. Commerce Department's Technology Administration. One objective of the NIST is to provide advice to federal agencies on interface standards for use in the acquisition of an ISEE (aka SEE). NIST is interested in uncovering gaps not covered by current standards efforts and in encouraging industry standardization efforts to fill those gaps. The NIST ISEE effort focused on SEE frameworks (see paragraph G.1.1.1) and the role repository standards, such as PCTE, fill in supporting SEE frameworks.

### **G.2.2 The North American PCTE Initiative (NAPI)**

NAPI (pronounced "nappy") grew out of a desire by the DoD and NIST to coordinate and promote interest among industry, academia, and government in the PCTE standard and in its implementation [CARNEY 93]. NAPI is currently (1994) being brought under the OMG.

### **G.2.3 International Standards Organization (ISO)**

ISO is a worldwide federation of national standards bodies. ISO develops standards through the use of technical committees, which may break down further into subcommittees and working groups. ISO is responsible for the Information Resource Dictionary System (IRDS) standards and for ISO 9000-3 certification of software development organizations [ROCK-EVANS 93a].

#### **G.2.4 American National Standards Institute (ANSI)**

ANSI is an official standards-making body. Through its accredited technical committees, it is working on a number of SEE-related standards. The technical committees of interest are as follows:

- X3H2 is developing extensions to the Structured Query Language (SQL).
- X3H4 is concerned with improving PCTE.
- X3H6 is investigating CASE tool integration models.
- X3H7 is evaluating object technology usage across standards organization.

#### **G.2.5 Institute of Electrical and Electronics Engineers (IEEE)**

IEEE is responsible for issuing a number of SEE-related standards. One of particular interest is the CASE tools interconnection standard known as P1175 [ROCK-EVANS 93a].

#### **G.2.6 European Computer Manufacturers Association (ECMA)**

ECMA was entrusted with the standardization and stabilization of the PCTE as a PTI.

#### **G.2.7 DoD's Ada Joint Program Office (AJPO)**

The DoD has sponsored the development of standards through a number of efforts, one of which is the AJPO. See CAIS-A in paragraph G.1.1.4.

#### **G.2.8 Object Management Group (OMG)**

The OMG is a nonprofit organization of almost 300 members, comprising an international community of software vendors, that focuses on standards-building for object-oriented technology. It uses a network of technical committees, subcommittees, task forces, and special interest groups whose members are drawn from member companies to do its work. See paragraph G.1.1.6.

### **G.2.9 Electronics Industries Association (EIA)**

The EIA produced a family of interim tool interchange standards collectively known as the CDIF.

### **G.2.10 Common Open Software Environment (COSE) Group**

COSE was formed in early 1993 in an effort to bring together companies interested in the development of a common UNIX standard. The principal companies are IBM, Hewlett-Packard, SunSoft, The Santa Cruz Operation, and Novell. One of the first fruits of the group was the CDE [LAWTON 93].

### **G.2.11 CASE Integration Services (CIS) Committee**

The CIS Committee is an industry consortium of hardware and software vendors and users; it addresses issues of CASE tool integration services using an object-oriented approach based on the ATIS. In late 1991 and early 1992, the committee's work was assumed by the creation of the ANSI Technical Committee X3H6 (See ANSI above) [SHU 92].

### **G.2.12 X/Open Company, Limited**

X/Open is a consortium of vendors promoting the practical implementation of open systems.

## **Appendix H: Bibliography**

[BRASHARES 93]

Brashares, D.C., *Application Portability and POSIX*, Fifth Annual Software Technology Conference, Apr 1993.

[BROWN 92]

Brown, A., A. Earl and J. McDermid, *SOFTWARE ENGINEERING ENVIRONMENTS: Automated Support for Software Engineering*, McGraw-Hill, 1992.

[CARNEY 93]

Carney, D.J., *The North American PCTE Initiative (NAPI)* (briefing charts), Software Engineering Institute, 1 Feb 1993.

[CURTIS]

Curtis, B., *Software Process Improvement Seminar for Senior Executives* (course charts), Software Engineering Institute, undated.

[CUTHILL 93]

Cuthill, B.B., *NISTIR 5295, Guide to Software Engineering Environment Assessment and Evaluation*, National Institute of Standards and Technology, Nov 1993.

[CUTHILL 94]

Cuthill, B.B., Comments provided to the STSC, Feb 1994.

[DRUYUN 93]

Druyun, D.A., "Software Reuse Incentive Policy," *CrossTalk*, Jan 1994: 5.

[FEILER 92]

Feiler, P. and W. Humphrey, *Software Process Development and Enactment: Concepts and Definitions* (SEI-92-TR-004), Software Engineering Institute, Sep 1992.

[FOWLER 90]

Fowler, P. and S. Rifkin, *Software Engineering Process Group Guide* (Technical Report CMU/SEI-90-TR-24), Software Engineering Institute, Sep 1990.

[HOROWITZ 91]

Horowitz, B., *The Importance of Architecture in DoD Software* (M91-35), The Mitre Corp., July 1991.

[HUMPHREY 90]

Humphrey, W.S., *Managing the Software Process*, Addison-Wesley Publishing Company, 1990 (reprint with corrections).

[IEEE 90]

*IEEE Std. 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology*, Standards Coordinating Committee of the IEEE Computer Society, 10 Dec 1990 (corrected 22 Jan 1991).

[ISSI 93]

*ProSLCSE User Documentation Vol. 2 Self Paced Tutorial* (ISSI-B91A00018A), International Software Systems, Inc., 6 May 1993.

[KARRER 90]

Karrar, A. and M. Penedo, *Software Engineering Environments Standard/Consensus Groups/Products—a summary* (Arcadia-TRW-90-002), TRW, 15 Aug 1990.

[LAWTON 93]

Lawton, G., "Cose Closes In," *Software Magazine*, Dec 1993: 28.

[MCCLURE 89]

McClure, C., *CASE Is Software Automation*, Prentice Hall, 1989.

[MCKEEHAN 93]

McKeehan, D., *The Capability Maturity Model: a Tutorial*, Software Engineering Institute, undated, (class given at STSC, June 1993).

[MOSEMAN 93]

Mosemann, L., *Lloyd K. Mosemann Speech at 1993 Software Technology Conference Salt Lake City, Utah on 22 April 1993* (SAF/PAS document 98-0565).

[NIST 91]

*NIST ISEE Glossary Version 1.0* (Draft), NIST ISEE Working Group, Apr 1991

[NIST 93]

*NIST Special Publication 500-211, Reference Model For Frameworks of Software Engineering Environments* (Technical Report ECMA TR/55, 3rd ed.), National Institute of Standards and Technology, Aug 1993.

[NIST 93a]

*Draft Guide to Software Engineering Environment Assessment and Evaluation*, National Institute of Standards and Technology, June 1993 (See [CUTHILL 93]).

[NGCR 93a]

*Report on the Progress of the NGCR PSESWG, Version 0.9*, Naval Air Warfare Center, Aircraft Division Warminster, 30 July 1993.

[NGCR 93b]

*Reference Model For Project Support Environments, Version 2.0 (Draft)*, Next Generation Computer Resources, 2 Sep 1993.

[PAULK 93]

Paulk, M.C., et al, *Capability Maturity Model for Software, Version 1.1* (CMU/SEI-93-TR-24), Software Engineering Institute, Feb 1993.

[PSESWG 93]

Project Support Environment Standards Working Group, *Reference Model For Project Support Environments, Version 1.0* (Report No. NAWCADWAR-93023-70), Naval Air Warfare Center, Aircraft Division Warminster, 28 Feb 1993.

[ROCK-EVANS 93a]

Rock-Evans, R., *Repositories and Frameworks: a Detailed Product Evaluation, Vol. 1*, Ovum Ltd., 1993.

[ROME 93]

*The Process-Oriented Software Life Cycle Support Environment (ProSLCSE)*, Rome Laboratory, 23 June 1993.

[SEI 93]

*CASE Environments: Integration and Adoption* (briefing charts), Software Engineering Institute, Apr 1993.

[SHU 92]

Shu, C., *PCTE and ATIS Comparison—A Summary* (Technical Report Arcadia-TRW-92-03), TRW, July 1992.

[STSC 92]

*Software Management Guide*, Software Technology Support Center, Oct 1990, revised Apr 1992.

[STSC 92a]

*Requirements Analysis & Design Tools Report*, Software Technology Support Center, Apr 1992.

[STSC 93]

*Test Preparation, Execution, & Software Technologies Report*, Software Technology Support Center, Feb 1993.

[WAKEMAN 93]

Wakeman, L. and J. Jowett, *PCTE the Standard For Open Repositories*, Prentice Hall, 1993.

[WHITE 93]

White, I., *The Booch Method: A Case Study for Rational Rose*, Rational, 1993.

(This page is intentionally blank.)

## **Appendix I: Recommended Reading**

1. *IEEE Std. 1209-1992, IEEE Recommended Practice for the Evaluation and Selection of CASE Tools*, IEEE Standards Board, 1993.
2. *NIST Special Publication 500-211, Reference Model For Frameworks of Software Engineering Environments* (Technical Report ECMA TR/55, 3rd ed.), National Institute of Standards and Technology, August 1993.
3. *PCTE the Standard For Open Repositories*, L. Wakeman & J. Jowett, Prentice Hall, 1993.
4. *Reference Model For Project Support Environments, Version 2.0* (Draft), Next Generation Computer Resources, 2 September 1993.
5. *Software Engineering Environments: Automated Support for Software Engineering*, A. Brown, A. Earl, & J. McDermid, McGraw-Hill Book Company, 1992.

The following list are reports that are available from the Software Technology Support Center:

1. *Requirements Analysis & Design Tools Report*, O. Bowler, et al., STSC, April 1992.
2. *Software Documentation Tool Report (Documentation Technology Report)*, G. Petersen & R. Sorensen; STSC, March 1993.
3. *Software Management Guide*, STSC, (3rd printing), April 1992.
4. *Test Preparation, Execution, & Evaluation Software Technologies Report*, G. Daich & G. Price; STSC, February 1993.